

**Univerzita Karlova**

**Filozofická fakulta**

**Ústav informačních studií a knihovnictví**

**Studijní obor: Informační studia a knihovnictví (7201T001)**

**Studijní program: Informační studia a knihovnictví (N7201)**

# **Diplomová práce**

**Bc. Sofya Garmash**

## **Kritická reflexe dopadů implementace agilních metod v projektování – případová studie ve vybraných společnostech v komerčním sektoru**

**The critical reflection of agile project management methods implementation —  
a case study in selected enterprises in the commercial sector**

**Praha 2018**

**Vedoucí práce: PhDr. Helena Lipková, Ph.D.**

Vedoucí diplomové práce: PhDr. Helena Lipková, Ph.D.

Oponent diplomové práce:

Datum obhajoby:

Hodnocení:

Prohlášení:

Prohlašuji, že jsem diplomovou práci vypracovala samostatně, že jsem řádně citovala všechny použité prameny a literaturu a že práce nebyla využita v rámci jiného vysokoškolského studia či k získání jiného nebo stejného titulu.

V Praze dne

.....

Podpis studenta

## **Poděkování**

Ráda bych poděkovala vedoucí práce PhDr. Heleně Lipkové, Ph.D., za její vstřícnost, trpělivost a cenné připomínky, stejně jako i respondentům, kteří se mého výzkumu zúčastnili. Další poděkování patří mé rodině, přátelům a kolegům, bez jejichž podpory by tato práce nevznikla.

## **Abstrakt**

Cílem diplomové práce je analyzovat problematiku implementace agilních metod a zmapovat pozitivní a negativní dopady tohoto procesu. První kapitola stručně popisuje historii a východiska dané skupiny metod řízení projektů, a to i s porovnáním agilních metod – s waterfallem a leanem. V druhé kapitole je pojednáno o principech agilních metod, jejich struktuře a zároveň o některých konkrétních metodách. Poslední část dané kapitoly se zaměřuje na význam agilních metod a jejich využití. Třetí kapitola zkoumá studie k danému tématu jak z českého prostředí, tak i ze zahraničí. Na základě provedené literární rešerše je vytvořena praktická část diplomové práce, která představuje kvalitativní výzkum ve třech společnostech a analýzu výsledků. Informace získané v průběhu polostrukturovaných rozhovorů a výsledky literární rešerše posloužily jako podklad pro doporučení, která jsou předložena v závěru práce.

## **Klíčová slova**

agilní metody, projektový management, kvalitativní výzkum

## **Abstract**

The goal of this work is to analyze the problems of implementation of agile methods and to map its positive and negative consequences. The first chapter briefly describes the history and sources of the given group of project development methodologies, including comparison of agile, waterfall and lean methods. In the second chapter, there is information about principles of agile methods, their structure and some concrete methods. The last part of the second chapter aims at the significance of agile methods and their use. The third chapter explores the studies of the given topic in Czech environment as well as outside of it. The practical part of thesis is based on the literature review and it presents the qualitative research of three companies as well as the analysis of the results. Information that was acquired during the semi-structured interviews and results of the literature research were used as the base for recommendations suggested in the conclusion of this work.

## **Keywords:**

agile methods, project management, qualitative research

### **Identifikační záznam**

GARMASH, Sofya. *Kritická reflexe dopadů implementace agilních metod projektování – případová studie ve vybraných společnostech v komerčním sektoru*. Praha, 2018, 125 s. Diplomová práce. Univerzita Karlova, Filozofická fakulta, Ústav informačních studií a knihovnictví. Vedoucí diplomové práce: PhDr. Helena Lipková, Ph.D.

# Obsah

Seznam zkratek .....	10
Úvod.....	11
1. Historické kořeny a východiska agilních metod projektování .....	13
1.1 Agilní metody versus waterfall .....	15
1.2 Lean.....	17
2. Základní principy, struktura a srovnání s jinými přístupy a metodami .....	19
2.1 Principy agilních metod .....	19
2.2 Struktura agilu aneb základní slovník pojmů.....	21
2.2.1 Role v agilu.....	21
2.2.2 Události v agilu: sprint a meetingy.....	24
2.2.3 Nástroje agilu: artefakty a měření .....	26
2.3 Příklady agilních metod .....	29
2.3.1 Scrum.....	29
2.3.2 Kanban.....	30
2.3.3 Extrémní programování (XP) .....	30
2.4 Význam agilních metod a jejich využití.....	31
2.4.1 Problematika implementace agilních metod .....	33
2.4.2 Agile a statistické údaje.....	34
3. Současný stav výzkumu v oblasti kritické reflexe agilních metod projektování .....	37
3.1 České prostředí.....	37
3.2 Zahraniční výzkumy.....	39
4. Praktická část – kvalitativní výzkum v konkrétních společnostech .....	48
4.1 Metodologie a popis výzkumu .....	48
4.2 Analýzy rozhovorů.....	49
4.2.1 Okruh 1: Obecné (úvod).....	50
4.2.2 Okruh 2: Scrum .....	52
4.2.3 Okruh 3: Problémy při implementaci .....	54
4.2.4 Okruh 4. Dodržování principů agilu a praxe .....	57
4.2.5 Okruh 5: Výsledky implementací.....	64
4.2.6 Okruh 6: Hodnocení a doporučení .....	66



4.2.7 Frekvenční analýza textů .....	69
Závěr a doporučení .....	71
Seznam použitých zdrojů .....	73
Seznam obrázků .....	75
Seznam příloh .....	76
Přílohy .....	77

# Seznam zkratek

DOD	definition of done
DSDM	Dynamic Systems Development Method
GDPR	General Data Protection Regulation
ICT	informační a komunikační technologie
IoT	Internet of Things
LeSS	Large-Scale Scrum
Manifest	Manifest agilního programování
PDSA	Plan-Do-Study-Act
PM	Project Manager
PMI	Project Management Institute
PO	Product Owner
RAD	rapid application development
RPA	Robotic Process Automation
SAFe	Scaled Agile Framework
SM	Scrum Master
TDD	test-driven development
TFS	Team Foundation Server
TPS	Toyota Production System
XP	extrémní programování

# Úvod

Poslední desetiletí se frekventovaně vyskytuje termín „agilní metody“ během mnohých výzkumů či činností, které jsou spojené s informačními a komunikačními technologiemi (dále ICT). Stává se běžným jevem nebo i samozřejmostí, že znalost těchto metod je potřebná při uplatnění na dnešním trhu práce a při jednání s nejrůznějšími pracovními společnostmi, neboť agilní metody jsou využívány jak nadnárodními korporacemi (Oracle, Amazon atd.), tak i menšími podniky, přičemž na jejich profilování nezáleží: „agilními“ mohou být konzultantské společnosti, hardwarové firmy nebo reklamní agentury (ŠOCHOVÁ, 2014, s. 9).

Tato popularizace agilních metod a ta poměrně optimistická nálada, která vládne kolem nich, vede k tomu, že se v dnešní době mnohé firmy snaží stát „agilními“, avšak při pokusu zavést nebo částečně použít dané metody se některé podniky potkávají s překážkami spojenými s lidskými zdroji či s organizací procesu řízení projektů, protože se ukazuje, že agilní metody mají své specifické požadavky na veškeré vnitřní firemní procesy a firemní kulturu, a to není snadné vyplnit, ani dodržovat. Tyto metody však bývají přebírány bez hlubších analýz na vhodnost pro konkrétní potřeby organizace, což může mít poměrně vážné důsledky pro danou společnost. Tato problematická stránka agilních metod, stejně jako dopady jejich implementace, ať úspěšné nebo ne, jsou v poslední době stále častěji zkoumány.

Cílem diplomové práce je zmapovat současnou situaci kolem implementace agilních metod v českém komerčním sektoru, a to s důrazem na pozitivní a negativní dopady daného procesu. Záměrem je identifikovat, pokud je to možné, a zanalyzovat podněty vedoucí k zavedení agilních metod, popsat zkušenosti lidí z reálného prostředí a navrhnout následná doporučení na základě výsledků šetření a literatury k danému tématu.

Zatímco popis kvalitativního výzkumu, jeho výsledků a návrh doporučení je praktickou částí této diplomové práce, teoretická část pojednává o vzniku agilních metod, jejich základních principech, struktuře a podobnostech/rozdílech s jinými metodikami, jako např. waterfall.

Tato diplomová práce má celkem 142.648 znaků s mezerami (79 normostran). Použité zdroje jsou citované dle ISO 690:2010. Literatura v seznamu na konci práce je uspořádaná jmenně abecedně; pokud jsou příjmení autora a rok vydání stejné u několika publikací, rozlišují se podle malých liter abecedy. Je použit Harvardský styl citování (odkaz ve formě uvedení příjmení autora, roku vydání a podle potřeby – i čísla citovaných stránek do kulatých závorek).

Po seznamu literatury je možné nalézt čtyři přílohy, které zahrnují otázky k rozhovorům, jejichž analýza je základem praktické části diplomové práce, a přepisy těchto rozhovorů.

# 1. Historické kořeny a východiska agilních metod projektování

Před tím, než budou principy agilních metod a jejich techniky rozebrány, je nutné udělat historický exkurz do minulosti, protože bez pochopení toho, jak a proč to začalo, není možné zkoumat stav a využití těchto metod projektování v současnosti.

Dle Rigby (2016b) lze za začátek nebo první námět agilních metod považovat práce Francise Bacona, avšak za rozumnější vztažný bod autoři počítají 30. léta minulého století. Walter Shewhart, pracující pro Bell Labs, začal tehdy používat cyklus Plan-Do-Study-Act (PDSA), který je možné považovat za prototyp pozdějších iteračních – a také agilních – metod. Od Shewharta se tomu přístupu naučil Edwards Deming, kterého později najala Toyota pro trénink svých managerů (RIGBY, 2016b). Za nějakou dobu Toyota na základě získaných zkušeností vyvine známý Toyota Production System, který ovlivnil nejen lean development (viz 1.2), ale i filozofii agilu.

V 70. a 80. letech začal být hojně používán waterfall, který v roce 1970 rozvinul Royce (MENS, 2008, s. 1), avšak po čase se začaly uskutečňovat kroky k tomu, aby byl proces vývoje flexibilnější a bližší k zákazníkovi. V roce 1985 Tom Glib napsal článek „Evolutionary Delivery versus the „waterfall model“, ve kterém mimo jiného nabízel jiný, ne-waterfallový způsob dodávání produktu reálnému koncovému uživateli, a to za použití krátkých intervalů (ABBAS, 2008, s. 98).

V roce 1986 byl v Harvard Business Review za autorství Takeuchi a Nonaka publikován článek „The New Product Development Game“ (RIGBY, 2016b). Autoři na základě výzkumu ve společnostech jako Canon nebo Honda ukázali, že práce je úspěšnější, pokud do ní jsou zapojeni všichni členové týmu, a to od začátku až do konce vývoje produktu; to pojmenovali „rugby přístup“ kvůli asociacím se stejnojmennou hrou (RIGBY, 2016b). Tento článek je často zmiňován ve spojení se Scrumem, protože je považován za předchůdce čili za inspiraci Scrumu, na což poukazuje i sám Jeff Sutherland jako jeden ze zakladatelů této agilní metodiky (RIGBY, 2016a).

Rok 1986 byl významným i z druhého důvodu. Barry Boehm ve své práci „A Spiral model of software development and enhancement“ prezentoval model tzv. iterativních metod, který navrhoval používat k identifikaci a snižování rizik (AGILE ALLIANCE, 2018). Avšak dá se říct, že iterativní metody byly svým způsobem používány už v 50. letech v NASA a IBM

Federal Systems Divisions; v NASA běžel v období let 1961–1963 projekt Mercury na krátkodobých iteracích (ABBAS, 2008, s. 97).

Podobný způsob řízení projektů, který je blíží k samotným agilním metodám, vyvinul James Martin v roce 1991 (AGILE ALLIANCE, 2018). Jde o RAD, neboli rapid application development, a cílem této metody bylo eliminovat proces přípravy k projektu a co nejrychleji začít pracovat. Byl popsán i opakovací časový interval, což se podobá konceptu sprintu v agilním přístupu. Později se z RAD vyvinul DSDM – Dynamic Systems Development Method.

Ve stejném období, v 90. letech minulého století, vznikaly techniky, které jsou dnes spojené s extrémním programováním (dále také XP), např. párové programování nebo společné vlastnictví kódu (ale významný článek o použití XP pod vedením Kena Becka na půdě společnosti Chrysler byl napsán jen roce 1998). Také byl poprvé popsán koncept stand-up meetingu. V roce 1993 Jeff Sutherland vyvíjí Scrum, avšak byl to spíš obecný koncept, který se až za dva roky začal podobat tomu, co je dnes známo. Pojem „sprint“ byl představen pouze v roce 1995 Kenem Schwabem a později, ve stejný rok, představil Sutherland a Schwaber Scrum na OOPSLA konferenci, ale ani tohle nebyla konečná varianta dnes známé metody. (AGILE ALLIANCE, 2018)

Paralelně se ocitnul softwarový vývoj v krizi. Příčina byla v tom, že technologický vývoj, vývoj společností se zrychlil, kvůli čemuž trvalo dodávání softwaru natolik dlouho (do tří let, ale mohlo to trvat i déle, bylo to např. v letectví), že ho už občas na konci zákazník ani nepotřeboval – trh se znovu změnil a produkt už nebyl vyhovující (VARHOL, 2015). Klasické metody řízení projektů přestaly být vhodné a dostačující, kvůli čemuž začaly být na konci 80. a začátku 90. let postupně implementovány výše popsané metody a praktiky.

Dalším významným milníkem se stal rok 2001. 11. února se v americkém státu Utah setkalo 17 profesionálů považujících se za vývojáře-anarchisty, kteří se zabývali v průběhu třech dnů procesem vývoje softwaru se záměrem projednat nové a „lehčí“ metody (STOICA, 2016, s. 1). Později získalo setkání název „Snowbird meeting“ dle názvu lyžařského střediska, ve kterém se uskutečnilo, ale zajímavostí je, že to nebylo první setkání této skupiny – už se viděli v roce 2000 v Chicago a Anguille (VARHOL, 2015). Extrémní programování a Scrum už existovaly a byly pokládány za „lehké“ metodiky, ale vývojáři s tím spokojení nebyli. Ve výsledku Kent Beck (zastávce extrémního programování) spolu se svými kolegy, mezi které patřili i např. Jeff Sutherland a Ken Schwaber (zakladatelé metodiky Scrum) nebo Alistair Cockburn (zakladatel metodiky Crystal), napsal Manifest agilního programování (ang. *Manifesto for Agile Software*

*Development*). Autoři vysvětlili, že k napsání tohoto dokumentu je navedla jejich vlastní praxe při tvorbě a vývoji softwaru, a schválili tyto čtyři hodnoty<sup>1</sup>.

*„Jednotlivci a interakce před procesy a nástroji*

*Fungující software před vyčerpávající dokumentací*

*Spolupráce se zákazníkem před vyjednáváním o smlouvě*

*Reagování na změny před dodržováním plánu“* (BECK et al., 2001).

Název „agile“ byl převzat z jedné knihy, která se jmenuje „Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer“ a kterou jeden z přítomných před setkáním přečetl (RIGBY, 2016b).

Kromě těchto čtyř hodnot autoři v průběhu následujících měsíců definovali 12 principů agilního vývoje softwaru, kterými je doporučováno řídit se. Podle Manifestu každý projekt, který tyto principy dodržuje, může být označen za agilní (STOICA, 2016, s. 1).

Později byl založen Agile Alliance, nezisková organizace, která měla za cíl podporovat a propagovat principy Manifestů a agilu ve světě.

Od té doby, co byl Manifest zveřejněn na svých webových stránkách, lidé mohli dokument „podepsat“ – ponechat svá jména pod textem Manifestu, a to až do roku 2016. V červenci 2016 Cunningham, jeden ze sedmnácti autorů a provozovatel oficiálních stránek o Manifestu, tuto možnost vynechal, ale stejně, dle jeho slov, více než 20 tisíc lidí to v průběhu 15 let podepsalo. (NYCE, 2017)

Důvod, proč to Cunningham udělal, bylo, že se už dle jeho názoru stal Manifest historií (NYCE, 2017). Výše popsané události patří také ze současného hlediska už k dějinám softwarového vývoje. Dnes se agilní metody neustále šíří světem, avšak vedle toho jsou stejně používány již zmíněné waterfall a lean. Následující část kapitoly uvádí popis těchto metod a jejich srovnání s agilem.

## 1.1 Agilní metody versus waterfall

Agile byl vyvinut jako protiklad už výše zmíněnému waterfallu (jinak „vodopádový model“, řidší „kaskádový model“), jež je klasický sekvenční model řízení projektů. Principy daného přístupu bohatě ovlivnily rozvoj agilních metod.

---

<sup>1</sup> Text Manifestu na oficiálních stránkách je k dispozici jak v angličtině, tak i v jiných jazycích, v češtině také.

Waterfall patří k tzv. tradičním modelům a je asi nejstarší ve své skupině (jak bylo řečeno dříve, tento model byl vyvinut Roycem v roce 1970). Jde o metodu, která je vhodná pro používání ve velkých týmech pro rozsáhlé projekty, rovněž pokud jde o dodržování přesných termínů a fixního rozpočtu (MYSLÍN, 2016, s. 24).

Základními komponenty watefallu jsou (MYSLÍN, 2016, s. 25):

- Požadavky,
- analýza;
- návrh;
- implementace;
- testování;
- předání;
- provoz.

Každá fáze probíhá jen jednou (není opakovatelná) a má své pořadí v procesu, díky čemuž vždy víme, ve které fázi vývoje projektu se nacházíme a co bude následovat dále, čímž je možné waterfall označit za poměrně předvídatelný a ve svém pořádku jednoduchý model. Kromě toho do následující části projektu vstupujeme pouze po ukončení předchozí. (MYSLÍN, 2016, s. 25)

Výše popsané body charakterizující vodopádový model je možné vnímat jako jeho výhody. K plusům patří i to, že *„každý člen týmu je specializován v konkrétní činnosti, kterou dokonale ovládá, případně se v ní dále vzdělává“* (MYSLÍN, 2016, s. 23). Doplnkově jednoznačnost a jednoduchost této metody mohou být lákavé, avšak v dnešní době waterfallový přístup poměrně často už nestačí.

K nevýhodám waterfallu patří náročnost identifikace a opravy chyb – to nutí vývojáře vrátit se na začátek projektu a přepracovávat ho (MYSLÍN, 2016, s. 26). Komunikace je formální, celý proces je neustále kontrolován, navíc ve waterfallu se vyskytuje poměrně vysoká úroveň formalizace (což se promítá i v rozsáhlejší dokumentaci) a celou strukturu lze označit za byrokratickou (SERRADOR, 2015, s. 1042). Je nezbytné zmínit, že zákazník není zapojen do práce – uvidí už hotový produkt a nemá šanci v průběhu projektu nijak ovlivnit jeho vývoj. Celý model postrádá flexibilitu, která je dnes nezbytná.

Může se zdát, že se waterfall v dnešní době už nehodí na projektové řízení, avšak to není pravda. Stále jsou typy projektů, kde lze waterfall úspěšně uplatnit. Myslín (2016, s. 24) uvádí, že takové projekty mohou být ty, nad kterými pracuje několik týmů zároveň, nebo které nemají jednotné umístění (jsou geograficky rozdělené). Z pohledu smlouvy se waterfall bude hodit



k projektům typu „fixed time, fixed price“. Ale pokud natolik striktní podmínky nejsou, v dnešní době by mohla společnost zvážit možnost přejít na agile, který má v porovnání s waterfallem své výhody.

Komunikace se zákazníkem je jedním z pilířů agilních metod. Stoica (2016, s. 1) zdůrazňuje, že agilní metody byly vytvořeny jako reakce na potřeby vývojářů, kteří se stále častěji setkávali s rozmanitými požadavky ze strany zákazníka, což je problematické zvládat při práci dle waterfallu, který není natolik flexibilní a poskytuje jen omezené možnosti přizpůsobit se zákazníkovi. Pokud porovnáme waterfall a agile, uvidíme, že hlavní odlišení mezi nimi je i v přístupu ke změnám: waterfall dbá na dodržování přesného schématu a na minimální počet změn, zatímco ve flexibilních agilních metodách jsou ony naopak vítány. Proces opravy chyb, kontrola projektu a požadavků se dělá pravidelně, zatímco ve waterfallu se to uskutečňuje až na konci.

Důležitým odlišením agilu je i princip samoorganizace. V týmu, který je základní jednotkou agilních metod, nedostávají lidé úkoly, ale každý dělá to, co si vybere sám, zatímco v klasickém modelu každý konkrétní člověk dostává konkrétní úkol. Podobná úroveň svobody patří i k vedení dokumentace: agile neklade na tuto část řízení projektů tak velký důraz jak to je ve waterfallu. Avšak to neznamená, že agile nepotřebuje rozpracování dokumentace vůbec. Jde spíš o minimalizaci a eliminaci nepotřebné nebo zatěžující práce.

Jak je vidět, rozdílů mezi agilem a waterfallem je dost. Existuje i jiný přístup k řízení projektů, se kterým se dá agilní metoda porovnat, a to je lean. Předem je nutné zdůraznit, že postoj k leanu není jednotný: někdo považuje lean za agilní metodiku, někdo ji vnímá jako úplně samostatný způsob řízení projektů; dokonce, jak Stoica (2016, s. 5) cituje Jeffa Sutherlanda, jednoho z „otců“ Scrumu, „*Agile is only a subset of the LEAN philosophy*“. Význam leanu pro rozvoj agilu je nepopíratelný, kvůli čemuž je její stručný přehled nezbytný pro tuto diplomovou práci.

## 1.2 Lean

Lean čili *Lean Software Development* je metodou řízení, která je považovaná za soubor principů a metod, jenž věnuje svou pozornost identifikaci a následně eliminaci činností, které nemají žádnou hodnotu při vytváření produktu nebo služby (MYSLÍN, 2016, s. 45). Daná metoda byla zavedena japonskou firmou Toyota po válce, kdy nebylo dostatečné financování a prostředí se neustále měnilo, což potřebovalo určitou úroveň flexibility od firmy, a byla ovlivněna filozofií „muda“, která právě trvá na eliminaci nepotřebných činností (STOICA, 2016, s. 6). Termín „lean“ nabídli výzkumníci z MIT, kteří v 80. – 90. letech tuto metodiku zkoumali (RIGBY,

2016b). Dnes je lean uznán celosvětově a má celou řadu doplnění. Kromě toho je na leanu založena metoda Kanban, která stojí na pomezí samotného leanu a agilních metod (viz podkapitolu 2.3.3).

Stejně jako agile, lean není vhodný pro každý podnik nebo tým. Tato metodika, která byla vytvořena pro tradiční výrobu, může najít uplatnění tam, kde je zapotřebí zjednodušit procesy a zrychlit předávání produktu zákazníkovi.

Co se týká odlišení leanu od agilu, Stoica (2016, s. 5) poukazuje na nižší flexibilitu leanu a jiné specifické nástroje k řízení projektu: lean používá hypotézy, analýzy, rozhovory se zákazníkem atd.

Agilní metody se dá porovnat nejen s leanem nebo waterfallem. Uvnitř samotné agilní skupiny lze najít konkrétní metodiky (jako výše zmíněné Scrum nebo extrémní programování), které se liší mezi sebou a nabízejí svým uživatelům odlišné techniky a jiné vnímání procesu řízení projektů. Cílem následující kapitoly je představit základní agilní techniky a procesy, uvést role a artefakty a na konci porovnat některé vybrané agilní metodiky mezi sebou.

## 2. Základní principy, struktura a srovnání s jinými přístupy a metodami

*Agilní* znamená proměnlivý, dynamický, přizpůsobivý; samo to slovo ukazuje, jaké jsou v této diplomové práci popsány metody. Avšak je nutné si uvědomovat, že nejde o tvrdý, přesný návod na to, jak změnit systém řízení projektů v podniku. Jak zdůraznil Myslín (2016, s. 20), „**Metodika není náboženství, metodika není dogma**“<sup>23</sup>. Nejde o obyčejný standardizovaný model, jakým je např. ISO. V podstatě jde o nový způsob života – „*být agilní znamená žít agilní filosofií...<...>...agilně musíte být, agilně musíte myslet, agilně se chovat*“ (ŠOCHOVÁ, 2014, s. 13). Zároveň Stoica (2016, s. 3) zaznamenává, že „*until now it is still unclear where philosophy ends and methodology applies or when methodology is applied in the spirit of philosophy*“, a říká, že agile je spíše filozofie, zatímco Scrum, XP, Kanban atd. jsou metody (viz podkapitolu 2.3), alespoň z pohledu klasického procesu vývoje softwaru (STOICA, 2016, s. 8). Jakoukoliv metodiku je možné označit za návod, který specifikuje činnosti patřící do jednotlivých procesů, a uvádí, kdo za to, kdy a jakým způsobem nese zodpovědnost. Pojem „agile“ nebo „agilní metody“ lze v takovém kontextu chápat jako souborné označení pro podobný typ metod, které mají společné rysy a takové definované kategorie jako role, artefakty atd. (viz podkapitolu 2.2). Neexistuje jedna jediná agilní metoda (i když Scrum, nejpopulárnější agilní metoda, je často synonymickým označením celého agilu), nýbrž několik, které se liší nejen od takových klasických metod řízení jako waterfall, ale v něčem i mezi sebou. Podrobněji o tom bude pojednáno dále.

### 2.1 Principy agilních metod

O základních principech agilu bylo zmíněno už v předchozí kapitole v pasáži o Manifestu agilního programování. Z těch hodnot, o kterých psal Kent Beck spolu se svými kolegy, vyplývá, že důraz je kladen na komunikaci, koncový produkt, minimální (podle možností) dokumentaci, spokojenost zákazníka a schopnost rychle reagovat na změny. Při podrobnější analýze je možné udělat i další závěry.

Tým, ve kterém je stanovena aktivní spolupráce mezi jeho členy, má přednost dle Manifestu před týmem s lepšími nástroji, ale s menší interakcí; úspěch mnohdy záleží na lidském faktoru více než na nástrojích a procesech. Avšak to neznamená, že by první tým měl používat horší

---

<sup>2</sup> Tučně je tato věta zvýrazněná v souladu s originálem.

<sup>3</sup> V této diplomové práci termín „metody“ je používán jako preferovaný, i když se docela často v literatuře vyskytuje jako synonymum slovo „metodiky“, jaké je to vidět z dané citace.

nástroje; jde o to, že bez komunikace v týmu a kreativního přístupu nelze dosáhnout kvalitních výsledků.

Každý potenciální uživatel (zákazník) by preferoval používat produkt, mít možnost ho zkusit na vlastní kůži bez rizika být pohřben pod návalem dokumentace (ať je technická či jakákoliv jiná), což je docela logický a adekvátní požadavek. To samé platí i pro dokumentaci uvnitř podniku: lidé by jistě preferovali věnovat více času samotné práci než psaní velkého množství interní dokumentace, která je bezpochyby potřebná (alespoň pro zachování a předávání znalostí), ale v menším množství, než je to u klasických metod, a kromě toho často navíc není aktuální. Zkrátka agile dbá na tom, aby stupeň formalizace a byrokracie byl nižší, ale zároveň nevyžaduje úplné odmítnutí dokumentace, bez které se projekt neobejde.

Co se týká třetího bodu: „*spolupráce se zákazníkem před vyjednáváním smlouvy*“ (BECK, 2001), zde je kladen důraz na stálou interakci se zákazníkem, a to nejen při testování produktu či podepsání smlouvy. Stává se, že (potenciální) klient neví přesně, co by chtěl dostat od dodavatele na konci, což předem podepsaná smlouva nevyřeší. Kromě toho v ideálním případě by bylo dobré získávat feedback na konci každé fáze vývoje projektu – to by usnadnilo jak průběh práce, tak i kvalitu konečného produktu. Produkt by měl být průběžně dodáván zákazníkovi, aby on chápal, jak se projekt vyvíjí a jaké změny je nutné (potřebné) provést. Zákazník by měl být v podstatě vnímán jako člen týmu. Kvůli tomu je tak důležité detailně projednat obsah a cíl projektu před podepsáním smlouvy. (ŠOCHOVÁ, 2014, s. 37–38)

Poslední základní princip pojednává o reagování na změny. Agilní metody vyžadují, aby při vývoji projektu byl řízen tým nejen plánem projektu (který by měl být flexibilní), ale i měnícími se požadavky zákazníka, které se bezpochyby budou vyskytovat v průběhu práce. Podobný přístup má za cíl zlepšit kvalitu koncového produktu a dosáhnout vyšší míry spokojenosti zákazníka (což je nejvyšší prioritou agilu) – ale bez porušování časových limitů, které mohou být předepsány smlouvou.

Doplňkově základní principy agilu míní, že spolupracující lidé se budou setkávat každý den (pro výměnu informací a vyhodnocování své práce<sup>4</sup>) a budou nejen dobře motivovaní, ale i schopní pracovat bez průběžné kontroly ze strany nadřízených (klasická role ředitele týmu v agilu neexistuje; podrobněji o rolích v agilu viz 2.2.1), což vyžaduje specifickou zodpovědnost ze strany každého člena týmu, talent k samoorganizaci a může přinášet spoustu problémů pro kvalitní a včasné splnění projektu. Nicméně podle agilu je nutné dávat určitou

---

<sup>4</sup> Vyhodnocování a následné zlepšení práce, hledání způsobu zvýšit efektivitu je důsledkem vlivu leanu – viz podrobněji podkapitulu 1.2.

míru svobody a důvěry vybraným jedincům; tohle je vliv japonských kořenů agilní filozofie, která kromě jiného zavádí i tzv. princip kolektivní viny (za chyby zodpovídají všichni členové týmu); avšak daný princip není moc používán v Evropě kvůli odlišnostem mentalit. Čas a praxe může zlepšit komunikaci a koordinaci týmu a vést k celkovému zlepšení práce.

Problém lze očekávat i ze strany zákazníka, který ne vždy může být přítomný na setkáních týmu, což v extrémním případě vyžaduje jmenování plně kompetentního zástupce zákazníka. Podobné platí i pro vyšší vedení podniku – není žádná garance, že top-manažer nebo Product Owner (viz 2.2.1) bude mít čas na pravidelnou komunikaci s týmem.

Pro lepší pochopení toho, jak jsou agilní principy uplatněny v praxi, je nutné udělat přehled vnitřní struktury agilních metodik, což je obsahem následující části druhé kapitoly.

## 2.2 Struktura agilu aneb základní slovník pojmů

Dle Schwabera a Sutherlanda (2017, s. 3), Scrum, z hlediska kterého je zde všechno rozebráno, má takové prvky jako Scrum tým se svými rolemi, artefakty, událostmi a pravidly. Vzhledem k tomu, že filozofie agilu byla popsána dříve, bude pozornost v této části diplomové práce věnovaná třem zbývajícím prvkům.

### 2.2.1 Role v agilu

V agilu je důležité dávat pozor na komunikaci a být loajálním ke změnám, ale nejdůležitějším prvkem jsou lidé. Bez silného, motivovaného týmu, bez pochopení toho, jak je nutné spolupracovat podle daného přístupu k řízení projektů, nebude výsledek nikdy ani úspěšný, ani přijatelný. Mezitím, ačkoliv se zdá, že agilní metody jsou velice flexibilní a nezaújaté, uvnitř daného systému panuje docela přesné rozdělení rolí mezi členy týmu (jako Product Owner a Scrum Master) a na samotný tým je kladen velký důraz.

Existuje rozdělení jedinců, kteří jsou jakkoliv s produktem spjati, na tzv. Pigs a Chickens. Toto pojetí nemá žádnou negativní konotaci. Chickens jsou lidé, kteří mají spíš pasivní vztah k projektu, což jsou např. konzultanti nebo koncový uživatel. Pigs je označením pro pracovníky, kteří přímo pracují nad projektem a mají za výsledný produkt odpovědnost. V této části diplomové práce je pozornost věnovaná zástupcům poslední skupiny. (MYSLÍN, 2016, s. 57)

**Product Owner** (dále také PO) je člověk, který kontroluje vývoj práce, definuje vizi produktu, ale zároveň není ředitelem týmu (ŠOCHOVÁ, 2014, s. 33-34). Jedná se o člověka, který

zastupuje zákazníka, přitom to může být jak přímo zaměstnanec klientské firmy, tak i najatý externista, a stará se o to, aby požadavky a představy zákazníka ohledně produktu byly splněny (MYSLÍN, 2016, s. 61, s. 67). Dle Šochové a Kunce (2014, s. 34): „*obvykle se čas Product Ownera dělí tak, že je 80% času u zákazníka a 20% v týmu*“<sup>5</sup>. Je přítomen u některých meetingů (viz následující podkapitolu). Má pravomoc zrušit sprint ve výjimečných situacích. Plní následující funkce (MYSLÍN, 2016, s. 64; ŠOCHOVÁ, 2014, s. 33–34):

- nese primární odpovědnost za projekt a práci vývojového týmu;
- stanoví priority u funkcionalit;
- spravuje product backlog (viz 2.2.3), zabezpečuje jeho dostupnost a porozumění obsahu týmem;
- zajišťuje komunikaci se zákazníkem.

Product Owner je občas natolik přetížený, že nemůže nést plnou zodpovědnost za správnost svých řešení a stanovení prioritních úkolů a potřebuje pomoc. Pokud PO nebývá pravidelně přítomen např. z důvodu práce v různých zemích, může jmenovat svým zástupcem člověka, který má možnost spolupracovat s týmem častěji než on sám. Taková role se nazývá **Product Owner Proxy** (ŠOCHOVÁ, 2014, s. 39). PO Proxy není tak často používaná role, avšak může sloužit jako doplněk k práci PO. V extrémním případě, pokud Product Owner nezvládá své povinnosti a PO Proxy není přidělen týmu, může na sebe vzít některé pravomoci Scrum Master, i když to není jeho povinnost.

**Scrum Master** (dále také SM) je pracovník ovládající metodiku Scrum a tím pádem je zodpovědný za porozumění a aplikaci pravidel Scrumu. Má za úkol pomáhat členům týmu s komunikací (nejen mezi sebou, ale i při jednání s druhými částmi podniku), sledovat principy agilu a pomoci týmu dospět k závěrům (ŠOCHOVÁ, 2014, s. 31). Organizuje a moderuje jednotlivé meetingy, pomáhá s dokumentací, popřípadě řeší technické problémy (jako např. nefunkčnost internetového připojení) (MYSLÍN, 2016, s. 74). Avšak zároveň nemusí být technickým odborníkem ani managerem v tom klasickém pojetí: „*Scrum Master by tým neměl nijak přímo řídit ani organizovat, ani za tým rozhodovat*“ (ŠOCHOVÁ, 2014, s. 32). SM se snaží navést tým k tomu, aby našel řešení samostatně. Je možné říct, že plní roli psychologa. Nesoudí strany, nezasahuje přímo do práce nad projektem, ale je vnímavým mediátorem, což vyžaduje odolnost vůči stresu a tvrdé nervy, obzvláště když má pracovat více než s jedním týmem. Motivuje tým k co nejlepším výsledkům a dozírá, aby nebyl tým zbytečně vyrušován.

---

<sup>5</sup> Tučně je tato věta zvýrazněná v souladu s originálem.

Dohlíží, aby každý v týmu (PO také) plnil svoji práci, zajišťuje potřebné věci a měří proces vývoje produktu za pomoci např. burndown diagramu (viz 2.2.3), (MYSLÍN, 2016, s. 73).

Důležité je to, že Scrum Master a Product Owner nesdílejí stejné funkce, spíš se vzájemně doplňují. Nesmí nastat situace, ve které jeden z nich bude vyplňovat práci druhého (ŠOCHOVÁ, 2014, s. 32).

Klasická role manažera ve Scrumu není předpokládána, avšak to neznamená, že by neexistovala vůbec. Jak zdůrazňují Šochová a Kunc (2014, s. 40), jde spíš o posouvání na vyšší úroveň – manager ve Scrumu má na starosti strategické rozhodnutí a pomáhá týmu, ale nekontroluje jeho práci. Zároveň může nastat situace, že se z klasického manažera stává buď Scrum Master, nebo Product Owner.

**Scrumový tým** (někdy také self-organized tým) se sestává ze specialistů, kteří jsou zodpovědní za vytvoření části projektů ke konci sprintu (viz následující podkapitulu). Hlavní funkci týmu tvoří implementace user stories (viz 2.2.3). Odlišností takového týmu je vzájemná důvěra, podpora, stejná vize produktu a společný cíl (ŠOCHOVÁ, 2014, s. 35). Navíc by v ideálním případě měli být členové týmu navzájem zastupitelní v tom, co se týká obsahu práce; např., tester může programovat.

Jak už bylo zmíněné výš, tým je samoorganizovaný, vyrovnaný, motivovaný a přesně ví (ne bez pomoci PO a SM), co dělat. Tým je celostní, což znamená v daném případě, že podtýmy nejsou vytvořeny (to se často stává během práce podle klasických metod řízení projektů); v podstatě jde o to, že každý člen týmu plní stejnou funkci, což je velmi důležité při dodržování metodiky Scrum (ŠOCHOVÁ, 2014, s. 35). To samé platí i pro zodpovědnost – občas je používán dříve zmíněný princip kolektivní viny, ale dost dobře to nefunguje. Stejná dvojice autorů v souvislosti s tím, co se týká zodpovědnosti, varuje: „*Pozor, nezaměňovat s anarchií, i v self-organized týmech vždy platí, že změnit lze pouze něco = existují hranice a procesy, které ani tento tým nemůže změnit*“ (ŠOCHOVÁ, 2014, s. 28).

Velikost týmu je stanovena podle množství práce a toho, nakolik rozsáhlé znalosti jsou potřebné. Více než např. devět lidí může být problémem, protože začne vyplňování stejné práce nebo rozptýl na nepotřebné činnosti. Také by bylo dobré, kdyby všichni členové týmu seděli v jedné místnosti.

Myslín vyjadřuje myšlenku (2016, s. 79), že by bylo vhodné sestavit tým jak z juniorů, tak i ze seniorů, protože zaprvé ne každý úkol potřebuje dlouholeté zkušenosti, zadruhé takovým způsobem budou připraveni noví pracovníci pro další úkoly a profesionální rozvoj.

## 2.2.2 Události v agilu: sprint a meetingy

V týmech, které pracují podle agilu, je práce rozdělena na jednotlivé časové úseky, které jsou navíc doplněné o specifické události nezbytné pro organizaci a využití času daného na projekt.

**Sprint** neboli time boxing je časově se opakující omezená jednotka práce. Obvykle má fixní předem stanovenou délku; typicky jeden až čtyři týdny, obvykle dva až tři (ŠOCHOVÁ, 2014, s. 44). Jednou za rok může být tato délka změněná. Ke konci každého sprintu by měl být pro zákazníka připraven určitý kus práce, a to reálně fungující část programu. Samotný zákazník by tím pádem měl možnost software otestovat a následně říct, zda je nutné něco měnit nebo ne. V průběhu jednoho sprintu pracuje tým se sprint backlogem (viz 2.2.3), odkud každý jednotlivý člen týmu přebírá a plní úkoly. Každý z těchto úkolů má od Product Ownera předem definovanou prioritu.

Pokud není předem stanovený úkol během daného sprintu splněn, je zpravidla buď přemístěn na konec celého projektu, nebo je úplně odstraněn ze sprintu. Zde je nutné zmínit, že podle agilu je nutné dělat co nejvíc práce za co nejméně času. Není vyloučeno, že některé činnosti, které se zdají být užitečné, nejsou potřebné vůbec a mohou být z projektů vynechané (ale podobný přístup k projektu by měl být vnímán s určitou skepsí).

Podle agilu není žádoucí měnit délku sprintu, např. dodávat jeden či dva dny do sprintu – tím pádem je porušen průběh projektu z pohledu filozofie agilu. Šochová a Kuncová dodávají (2014, s. 44), že nikdo nemá pravomoc měnit obsah sprintu, ani Product Owner, v jeho průběhu.

Nezbytným pojmem je **daily stand-up (daily meeting, scrum meeting)**. Jedním z hlavních bodů agilu je spolupráce, a právě k tomu slouží dané setkávání týmu. Uskutečňuje se každodenně ráno ve stejnou dobu a ve stejném místě, trvá maximálně patnáct minut a probíhá ve stoje, což je ideální případ. Cílem je zlepšení komunikace a identifikace problémů, pokud nějaké jsou (MYSLÍN, 2016, s. 112). Každý člen týmu<sup>6</sup> informuje ostatní, co udělal včera a co bude dělat dnes (ale samotné řešení zde není projednáváno). Největší význam má daily stand-up, pokud členové týmu pracují na stejných věcech, ale častěji to bývá jinak.

O věcech, jako jsou délka sprintu a jiné důležité aspekty pro projekt, se diskutuje na **plánovacím meetingu**. Tým se seznamuje, plánuje se product backlog a diskutuje se ohledně celého projektu. Odhaduje se náročnost a cena, stanoví se akceptační kritéria a priority (za to zodpovídá Product Owner), diskutuje se o komunikačních kanálech a řeší se jiné

---

<sup>6</sup> Product Owner se daily stand-upu nezúčastní nebo je tam pouze jako pozorovatel (MYSLÍN, 2016, s. 113).



organizační záležitosti. Pokud je to zapotřebí, může být později proveden tzv. **Backlog Refinement Meeting**, na kterém product backlog nabývá finální podoby s pomocí Product Ownera, který upřesňuje požadavky zákazníka (MYSLÍN, 2016, s. 104).

Dle agilní metodiky by neměl plánovací meeting trvat déle než 4 hodiny, avšak pokud jde o měsíční sprint, čas může být prodloužen až do osmi hodin, ale v podobném případě je více žádoucí provedení groomingu. **Grooming (Backlog Grooming)** je typem meetingu, který předchází plánovacímu meetingu, pokud jde o velký tým nebo větší množství úkolů. Řeší se ocenění (stává se, že se mění priority, což ovlivňuje cenu), Product Owner odpovídá na dotazy týmu. Výstupem je podklad pro plánovací meeting. (ŠOCHOVÁ, 2014, s. 83)

Už po groomingu a plánovacím meetingu, který určuje průběh celého projektu, lze mluvit o jednotlivých sprintech. Každému z nich předchází **Sprint Planning Meeting**, který může trvat kolem dvou hodin. Určuje se, co bude implementováno, cíle daného sprintu<sup>7</sup>, identifikují se činnosti, které by měly být provedené. PO dělá sprint backlog<sup>8</sup> a akceptační kritéria ke každé user story (viz 2.2.3), paralelně vysvětluje kontext a obsah sprintu. Dále následuje diskuse, moderována SM, během které PO odpovídá dotazům ze strany týmu. (MYSLÍN, 2016, s. 108–109).

Na konci každého sprintu se koná **review** (jinak **Sprint Review Meeting**), což je prezentace výsledků pro zákazníka (MYSLÍN, 2016, s. 115). Prezentace není udělaná pomocí slajdů nebo jiným podobným způsobem – pouze naživo; důležité je, jak reálná aplikace vypadá *ve skutečnosti*. Ukazuje se pouze ten kousek projektu, který je dokončen (není vyloučeno, že si to zákazník bude chtít vyzkoušet), což jsou v podstatě user stories se splněnými akceptačními kritérii (ŠOCHOVÁ, 2014, s. 82). Ideální by bylo získat mínění nejen zákazníka, ale i konečného uživatele – to by pomohlo udělat kvalitu produktu ještě vyšší.

Může nastat situace, že se zákazník bude domnívat, že už mu stačí tahle ta podoba, a další sprint nebude proplácet (ŠOCHOVÁ, 2014, s. 38). K tomu patří tzv. **agilní smlouva** (effort-based spolupráce), kdy zákazník platí jen za jeden sprint a pak řeší, jestli bude ještě jeden zapotřebí (ŠOCHOVÁ, 2014, s. 117).

Dělají se připomínky, které jsou později vkládány do product backlogu<sup>9</sup>. Podle potřeby je upraven datum odevzdání projektu (ale to se nedělá po prvním sprintu, spíš po uběhnutí

---

<sup>7</sup> Není žádoucí, aby se cíle měnily v průběhu sprintu – to narušuje proces splnění úkolů a práci týmu.

<sup>8</sup> Později za to, jak se pracuje s backlogem, zodpovídá Scrum Master. Pokud proces nejde podle plánu, je vyžadována spolupráce s Product Ownerem.

<sup>9</sup> Podrobněji o tomto pojmu viz následující podkapitulu 2.2.3.

poloviny sprintů, pokud nebude produkt zjevně včas odevzdán). Diskutuje se nad dalším postupem (jaké úkoly se budou řešit během dalšího sprintu – podklady pro následující plánovací meeting), existujícími technickými problémy, co se nestihlo, co proběhlo dobře. Celkově může dané setkání trvat čtyři hodiny.

Potom se uskuteční **Sprint Retrospective Meeting**, zkratka **retrospektiva**. Shrnuje se, co proběhlo dobře či špatně; dbá se na to, co by mohlo být zrušeno nebo zlepšeno<sup>10</sup> (ŠOCHOVÁ, 2014, s. 28). Organizuje a moderuje to Scrum Master. Dokud se technická stránka projektu řeší spíš na review meetingu, jsou zde projednávány procesy, vztahy a nástroje; důležité je, aby se všichni členové týmu vyjádřili a bylo přijaté závěrečné rozhodnutí (MYSLÍN, 2016, s. 117).

Posledním setkáním je **konečný meeting**. Předává se dokumentace zákazníkovi a domlouvá se popřípadě na záruční lhůtě (MYSLÍN, 2016, s. 119). Počítá se s tím, že produkt byl otestován a přijat zákazníkem ve své finální podobě.

### 2.2.3 Nástroje agilu: artefakty a měření

Agilní metody používají speciální nástroje, mezi které patří product backlog, sprint a release backlog, různé aplikace. To všechno projektový tým v čele s Product Ownerem a Scrum Masterem potřebuje, aby dodal koncový produkt, což se stane poté, jakmile budou splněny požadavky zákazníka, a k tomu nezbytně patří pojem „user stories“.

**User story** neboli uživatelský příběh je scénář, popis toho, co se očekává od systému, že bude splňovat (MYSLÍN, 2016, s. 88). Nejde o to, jakým způsobem to bude prováděno – za to zodpovídají vývojáři z týmu. Pro zákazníka, který často mluví jiným jazykem, je důležitější výsledek, příliš ho nezajímá technická stránka. Pro vzájemné pochopení jsou používané právě user story – krátké jednoznačné věty, kde uživatel v jakési roli (např. administrátor) uvádí, co chce získat a za pomoci jaké činnosti (MYSLÍN, 2016, s. 88). User stories jsou založené na představě zákazníka a mají jednotnou podobu, avšak nejsou nijak přesně formalizovány. Např. user story může vypadat takto: „*Jako uživatel chci, abych mohl vyhledávat ostatní uživatele podle jména a příjmení*“ (MYSLÍN, 2016, s. 89). Definují se role (uživatel), cíl nebo specifikace celého user story („vyhledávat ostatní uživatele“); popřípadě užitek (abych měl něco na konci), ale poslední část není povinná (MYSLÍN, 2016, s. 89). Důležité je sestavit user stories pro všechny možné role a předvídat různé scénáře. Šochová a Kuncové doporučují (2014, s. 63) mít minimálně pět až deset user stories na sprint a aby největší

---

<sup>10</sup> Tuto snahu ovlivnil lean – viz 1.2.

user story nezabrala víc než polovinu onoho sprintu (dodržení granularity). Ideálně by měl člen týmu pracovat jen nad jednou úlohou.

Na sprint planning meetingu jsou pro každý user story připravená Product Ownerem akceptační kritéria. Jde o proces ověřování, testování, že na základě určité operace dostaneme předem definovaný výstup. Pokud jsou všechny podmínky – tato kritéria – splněny, user story byla vyplněna úspěšně a je považována za dokončenou. (ŠOCHOVÁ, 2014, s. 27)

**DOD (definition of done) čili done kritéria** – požadavky na úkol, který má být považován za splněný; liší se od akceptačních kritérií tím, že jsou globálnější a platí pro celý projekt. Příkladem může sloužit bod „úspěšné otestování (manuální)“ nebo „dokumentace ke každé funkcionalitě“ (ŠOCHOVÁ, 2014, s. 53). Je klíčovým aspektem Scrumu z hlediska zajištění kvality výstupů. DOD je stejný pro všechny úkoly. PO by měl dodržovat co nejvyšší standard DOD, ale občas nastává problém ve srozumitelnosti (např. PO nechápe všechny požadavky z pohledu potenciálního uživatele, pouze je kontroluje). Finální akceptaci potom dělá zákazník, samotné DOD nemůže být vnímáno podobným způsobem.

**Product backlog** je soubor úkolů čili seznam user stories, které je nutné v průběhu projektu do systému implementovat (MYSLÍN, 2016, s. 97–98). Prvotní odpovědnost za sestavení daného dokumentu nese PO; v průběhu rozvoje projektu se položky mohou měnit podle obsahu nebo dle priority (ŠOCHOVÁ, 2014, s. 27). Avšak Product Owner nemusí všechno dělat sám – při zkoumání prostředí mu mohou pomáhat konzultanti z takových oddělení, jako např. marketing. Nejčastěji má podobu seznamu nebo tabulky, kde je možné nalézt user stories s přiřazenými prioritami a stavem. Položka může mít jeden z těchto stavů: nová, rozpracovaná/in progress (někdo na tom začal pracovat), zablokovaná (přerušená kvůli nějakému závaznému důvodu) nebo dokončená/done (ŠOCHOVÁ, 2014, s. 55). Jestli se úkol nachází ve stadiu rozpracování, PO ho může nechat na další sprint.

Čím vyšší prioritu má položka, tím více je rozpracovaná a je vnímána jako podklad pro backlog sprintu, kvůli čemuž je důležité důkladné rozpracování vyskytujících se položek se zaměřením na budoucnost (ŠOCHOVÁ, 2014, s. 46). Položka s nižší prioritou není natolik podrobná a čeká na další analýzu.

**Sprint backlog** se zaměřuje na aktuální sprint; položky se převezmou z product backlogu (kde budou potom vymazané), ale pokud se její vyplnění nestíhá, tak položka (úkol) bude s největší pravděpodobností vrácená zpátky (ŠOCHOVÁ, 2014, s. 45). Na rozdíl od product backlogu sprint backlog v průběhu sprintu není možné měnit. Obsah sprint backlogu se plánuje na sprint

meetingu (viz 2.2.2) s takovým záměrem, aby přidané user stories do tohoto backlogu byly logické, souvislé mezi sebou, aby na konci sprintu byl dosažen předem stanovený cíl.

Všechno je napsáno tak, aby každý člen týmu viděl stav sprintu. Jako nástroj pro tento cíl je možné použít speciální software, např. se kupuje Assembla, Jira Agile, Confluence nebo i Stash; z open source je znám Openproject. Stoica (2016, s. 4) dodává k tomuto výčtu MS Project, Sprintometr, IBM Rational nebo HP Agile Manager. Vhodným způsobem vizualizace je metoda Kanban (viz 2.3.3). Nejjednodušší způsob práce s backlogem je obyčejná tabule v místnosti, kde tým pracuje.

Mezi nástroje agilu patří i způsoby měření úspěšnosti projektu. Za to odpovídá Scrum Master, který dohlíží, zda tým stihne zpracovat všechny úkoly v průběhu sprintu a jaká je velocity (rychlost týmu). Nejčastěji je k tomu používán už dříve zmíněný **burndown chart** (také **burndown diagram**<sup>11</sup>). Ten ukazuje, kolik úkolů z těch, co jsou propsané v product backlogu, zbývá, což může hodně pomoci Product Owneru při sledování týmové práce (MYSLÍN, 2016, s. 122). Tento chart doplnkově demonstruje, v jaké fázi sprintu se nacházíme a jestli to stíháme. Pro každý nový sprint se vytvoří nový chart, který je pro lepší viditelnost vhodné kreslit na tabuli přímo v té místnosti, kde tým pracuje, protože tohle je ten přehlednější nástroj (ŠOCHOVÁ, 2014, s. 45). Jako výhody burndown chartu Myslín (2016, s. 126) uvádí jednoduchost a přímý údaj, jako nevýhody – to, že úkoly mohou mít odlišnou náročnost a tím bude diagram zkreslen.

Na základě několika burndown diagramů lze vypočítat výše zmíněnou **velocity**, což je ukazatel střední týmové rychlosti (kolik úkolů je schopen za sprint udělat). Měří se v bodech. „*Slouží k lepšímu odhadu práce pro příští Sprints a je též ukazatelem efektivity týmu*“ (ŠOCHOVÁ, 2014, s. 28).

Objem práce je možné měřit v tzv. člověkohodinách (man/hours)/člověkodnech (man/days), nebo za pomoci metody s názvem **planning poker**. Dle Šochové a Kunce (2014, s. 27): „*obvykle se hraje ve formě herních karet s tzv. Fibonacciho řadou pro určení velikosti (komplexnosti) User Story*“. Pojetí komplexnosti nezahrnuje jen náročnost zpracování, ale i testování a patřičnou dokumentaci (ŠOCHOVÁ, 2014, s. 58). Jde o stupnici „0, ½, 1, 3, 5, 8, 13, 20, 40, 100, ?, nekonečno a kartu s kafem“ (ŠOCHOVÁ, 2014, s. 61). Nekonečno ukazuje na vysokou komplexnost, „?“ je pro nejistotu a nutnost zjistit víc a kafe je nemá žádost o přestávku. Každý člen týmu po výběru jedné karty vysvětluje ostatním v čele se Scrum

---

<sup>11</sup> Obvykle se tento termín nepřekládá do češtiny a používá se v originálním znění, avšak dá se přeložit jako „graf spalování (práce)“

Masterem, proč zvolil právě ji. Planning poker pomáhá zlepšit jak komunikaci, získat mínění každého člena týmu, tak i odhadnout přibližnou náročnost prací, což by mělo ušetřit čas a peníze v budoucnu.

Planning poker je technikou patřící ke Scrumu, stejně jako většina v této části diplomové práce popsaných artefaktů a událostí, ale to neznamená, že se agile omezuje pouze na tuto metodiku. Následující část kapitoly má za cíl ukázat, jaké další agilní metodiky existují a co je jejich podstatou.

## 2.3 Příklady agilních metod

Stává se, že se pod pojmem „agilní metodika“ míní pouze výše zmíněný Scrum, který je asi nejpoužívanějším mezi jinými svého druhu, avšak to neznamená, že je možné opomenout i jiné metody. Do agilních metod lze např. zařadit také extrémní programování (XP), feature-driven development (FDD), Crystal Clear, Kanban a takové „kompromisní“ hybridy, jako Scrumban nebo Scrum/XP (STOICA, 2016, s. 4). V této podkapitole je uveden stručný přehled třech vybraných agilních metodik: Scrumu, extrémního programování a Kanbanu.

### 2.3.1 Scrum

Metodika, kterou vyvíjel Jeff Sutherland spolu s Kenem Schwabem v 90 letech a která byla publikována ve své finální podobě v roce 1995. Ovládá dnes více než 50 % trhu, zatímco např. Kanban má 5 % (VersionOne, 2017).

Dle výše zmíněných autorů je Scrum „*lightweight, simple to understand, difficult to master*“ (SCHWABER, 2017, s. 3). Navíc zdůrazňují, že „*Scrum is not a process, technique, or definitive method. Rather, it is a framework within which you can employ various processes and techniques.*“ (SCHWABER, 2017, s. 3).

Obecná struktura dané metodiky již byla v práci popsána. Pouze stojí za zmínku že dle Schwabera a Sutherlanda (2017, s. 4–5) stojí Scrum na třech pilířích: „*transparency, inspection, and adaptation*“.

**Transparentnost** znamená, že cíl je jasně definován a všichni ho chápou stejným způsobem. Lidé rozumí, co a proč dělají, používají společné normy a jednotnou terminologii.

**Kontrola (inspekce)** výkonu práce (čili její pokrok) a artefaktů je o eliminaci odchylek i minimalizaci chyb. Problémem se může stát dosažení rozumného kompromisu mezi tvrdou kontrolou, která je příznačná spíše pro klasické metody typu watefall, a úplnou absencí revize.

Podle agilu by měl být tým schopen pracovat samostatně, ale na začátku je nezbytné provádět inspekci. Tím pádem nejlepším řešením je koučování, které eliminuje pocit, že jsou totálně hlídání a kontrolování. Scrum dbá na to, aby byl kompromis dodržen.

**Adaptace** je následkem inspekce. Pokud bylo zjištěno, že odchylka proběhla a produkt může nevyhovovat zákazníkovi, proces je nutné upravit, aby se situace nezopakovala. K tomu slouží události, popsané v 2.2.2.

### 2.3.2 Kanban

Šochová a Kunc (2014, s. 27) charakterizují Kanban jako metodiku „*na pomezí Agile a Lean fungující na principech vizualizace procesu, omezení rozpracované práce a minimalizace času potřebného na dokončení jednotlivých celků*“. Někteří se domnívají, že tohle je spíš způsob organizace práce než zvláštní metodika. Stoica (2016, s. 6) podtrhuje, že Kanban je více praktický než Scrum a má technický, průmyslový charakter, což vysvětluje vlivem leanu a produkčním systémem Toyota (TPS – Toyota Production System). Jak bylo výše uvedeno, Kanban je hlavně o vizualizaci procesů, na to se používají barevné štítky (nejjednodušší varianta) nebo i speciální software. Také jde o zlepšení komunikace, monitorování toho, jak jsou vyplňovány úkoly, a optimalizaci celého procesu – dělat jen to, co je teď potřebné, a začínat nový úkol až po skončení předchozího (STOICA, 2016, s. 6; ŠOCHOVÁ, 2014, s. 105).

Scrumban, zmíněný v podkapitole 2.3, je hybridním spojením této metodiky a Scrumu. Dle výzkumu VersionOne (2017) je Scrumban používán u 8 % dotazovaných.

### 2.3.3 Extrémní programování (XP)

Název této metodiky je takový, protože „*dovádí do extrému mnohé principy známé z jiných metodik*“ (MYSLÍN, 2016, s. 41). Pokud je, např. ve Scrumu, sprintu věnováno několik týdnů, v extrémním programování jde pouze o pár dní (nebo i o jeden den), přičemž přesčasy nejsou vítány. Testování také neprobíhá jednou za nějaké období, ale pořád, přičemž čím dříve proces začne, tím lépe; tato praktika je spojená s test-driven development, jinak TDD (ŠOCHOVÁ, 2014, s. 27). Zvláštní technikou XP je „*parní programování*“, kdy dva programátoři pracují spolu u jednoho počítače. Jeden píše kód, druhý proces sleduje, popřípadě komentuje a upravuje. K jiným vlastnostem XP lze zařadit např. standardizace (aby každý mohl pochopit, co napsal kolega) a společné vlastnictví (kód může upravovat kdokoliv z týmu); jinak praktik v extrémním programování je mnohem víc. Co se týká změn, mohou být implementovány

několikrát denně, a to je možná „nejextrémnější“ bod celé metodiky. Myslín uvádí (2016, s. 42), že k základním hodnotám extrémního programování patří komunikace, jednoduchost, zpětná vazba a odvaha. Komunikace v XP je taková, že u procesu vývoje je neustále přítomen i zákazník a členové týmu mají pořád komunikovat nejen mezi sebou, ale i s ním pro zajištění lepšího výsledku. Jednoduchostí je myšleno, že se v průběhu prací nad projektem dělá pouze to, co je opravdu zapotřebí. Zpětná vazba o tom říká, že se kód nepřetržitě testuje, aby změny bylo možné udělat ihned. Poslední bod, odvaha, znamená, že při takových pracovních podmínkách je nezbytné mít kuráž pro rychlé rozhodování (zde nikdo čekat nemůže) nebo pro vymazání půlky kódu, s kterým bylo odvedeno mnoho práce (ŠOCHOVÁ, 2014, s. 102).

Techniky extrémního programování jsou populární při práci s jakoukoliv agilní metodikou. Např. dle výzkumu VersionOne (2017, s. 11) se výše zmíněná standardizace používá u 74 % respondentů, zatímco test-driven development používá 40 %, párové programování – 32 %. Vedle toho vyvstává otázka, jak a kde je vhodné XP nebo obecně agilní metody používat a proč je tomu dnes nutné věnovat pozornost.

## 2.4 Význam agilních metod a jejich využití

V současné době, kdy se technologie mění ještě rychleji, než tomu bylo v 80. nebo 90. letech, je schopnost být flexibilním a „agilním“, bez velkých problémů zvládat změny, cenná jako nikdy předtím. Flexibility a změny jsou v dnešní době ostrou a aktuální problematikou. Inovace i trendy hýbají světem, to má bezpodmínečně vliv jak na zákazníka a jeho požadavky či vidění produktu, tak i na naši konkurenceschopnost – podniky se pořád potýkají s tlakem zevnějšku a musejí reagovat na to, jak se mění podmínky vedení businessu. Tak se stalo, že se rychle měnící svět kolem nás i neustálý technologický pokrok ovlivňuje i náš pracovní postup, ať už je to žádoucí nebo ne. Tím pádem je vyžadováno měnit se a umět se přizpůsobit každé nové výzvě nebo změně. Jsou to vlastnosti, které jsou dnes nezbytné pro přežití podniku na trhu. Tím pádem tato skupina metodik nabývá obrovského významu nejen pro vývoj softwaru, ale i pro jakékoliv business procesy.

Agilní metody jsou nejen o flexibilitě a změnách. Jsou i o inovacích. Rychlý vývoj současných technologií a měnící se životní prostředí vyžadují zcela nový přístup jak k vývoji softwaru, tak i k jakýmkoliv jiným operacím ohledně výroby služeb a produktů. Tato skutečnost a neustálý pokrok nutí společnosti zavádět inovační technologii co nejrychleji – pro dodržování vysoké úrovně konkurenceschopnosti, a právě agilní metody jsou v tom velice nápomocné. O tom zmiňují Rigby, Sutherland a Takeuchi (2016a), věnuje se tomu Sutherland i s Kenem

Schwaberm ve svém tzv. *Průvodce Scrumem: Pravidla hry*<sup>12</sup>, jenž je základní příručkou pro ty, kteří mají o Scrum zájem. Ve svém Průvodci Scrumem Schwaber a Sutherland (2017) kromě jiného rozepisují, k čemu může být Scrum používán. Jde nejen o výzkum trhu, vývoj a zdokonalení produktu, ale i např. o update produktu několikrát za den nebo o vývoj a podporu cloudových technologií (SCHWABER, 2017, s. 4). Vzhledem k tomu, že poslední roky právě cloud computing nebo Internet of Things (IoT) patří k technologickým trendům, Scrum zjevně může pomoci firmám být nejen „trendovými“ v tom, co se týká používání agilu, ale také pomůže rozšířit své možnosti v oblasti cloudových technologií.

Na jiné benefity poukazují i Rigby, Sutherland a Takeuchi (2016a). Uvádí dlouhý výčet přínosů agilních metod, mezi které patří spokojenost pracovníků a jejich výkon, rozšíření zkušeností mezi těmito lidmi v rámci týmu a prohloubení jejich vzájemné důvěry a respektu. Agile pomáhá minimalizovat plýtvání, které může vzniknout např. kvůli opakovacímu plánování, přebytké dokumentaci nebo závadám zboží. Protože jsou zákaznické priority na prvním místě, probíhá neustále adaptace, zlepšuje se komunikace se zákazníkem, a tím se zvyšuje jeho následná spokojenost, zrychluje se proces dodávání produktu na trh a eliminují se rizika s tím spojené. Agilní přístup může vychovat managery, kteří budou v budoucnu sami flexibilnější, budou mít blíž zákazníkovi, a to jsou rysy, kterých je dnes nedostatek. Nakonec, kvůli změně paradigmatu řízení projektů, se mění i role projektového manažera, který se může víc věnovat samotné společnosti, její strategii a vizi. (RIGBY, 2016a)

Stejní autoři poukazují nejen na výhody agilních metod, ale i na to, kde mohou být používány. Příkladem, že se agilní metody šíří světem a napříč odvětvími, může být to, že je lze využít nejen pro vývoj softwaru, se kterým především je agile spojen, jak uvádí autoři National Public Radio, Saab, General Electric, ale dokonce i pro vinařství Mission Bell, které „uses them for everything from wine production to warehousing to running its senior leadership group“ (RIGBY, 2016a). Z pohledu autorů (RIGBY, 2016a) mohou agilní metody nejen zrychlit růst zisku, ale také vytváří novou generaci managerů, kteří budou mít dovednosti na vysoké úrovni. Navíc agile otevírá nové perspektivy pro taková oddělení jako marketing (lépe lze říct – individuální práce s každým klientem), human resources (pracovníci, kteří mají motivaci), poradenství nebo řízení dodávek (RIGBY, 2016a). Agilní metody je možné používat i mimo softwarový vývoj nebo IT oddělení.

---

<sup>12</sup> V této diplomové práci je citována poslední verze Průvodce (z listopadu 2017), avšak překlad do češtiny není k dispozici, proto je používána originální anglická verze.



Podobně se vyjadřují i Ken Schwaber a Jeff Sutherland; píšou, že se dá implementovat Scrum v různých oblastech, a to např. „*schools, government, marketing, managing the operation of organizations*“ (SCHWABER, 2017, s. 4). Zároveň autoři (SCHWABER, 2017, s. 4) mluví o tom, že se tuto metodiku dá využívat každodenně k řízení skoro čehokoliv.

Šochová a Kunce ihned na začátku své knihy „Agilní metody řízení projektů“ (2016) poukazují na to, že tyto metody jsou používány takovými společnostmi, jako Salesforce a Google, a jako příklad podniků, kde by se dalo agile uplatnit, uvádí dlouhý výčet: „...*najdete je na univerzitách, v telekomunikacích, v bankách a pojišťovnách, v armádě, aerolinkách, automobilovém průmyslu, medicínských firmách a samozřejmě i v mnoha malých či středních firmách*“ (ŠOCHOVÁ, 2014, s. 9). Tak to mají Spotify a Twitter a experimentoval s tím Walmart. Jako i předchozí autoři podtrhují, že agilní metody mohou být používány kdekoliv, kde je nutné kladt důraz na flexibilitu a změny.

Z výše uvedených bodů k významu a z popisu toho, kde je možné agilní metody využít, je patrně vidět, že při vhodném nasazení a úspěšné implementaci mají spoustu výhod a přinesou společnosti velký počet benefitů a zvýší její konkurenceschopnost. Avšak podobné změny ve firmě vyžadují zvýšenou opatrnost a pochopení, proč se to dělá a co tím firma chce získat. Jinak se vedení firmy může setkat s problémy, které jen zhorší stávající situaci. O tom krátce pojednává další část kapitoly.

#### 2.4.1 Problematika implementace agilních metod

Je nutné zdůraznit, že ne každý projekt je vhodný na to, aby na něm byl použit agilní přístup – někdy je lépe dát přednost waterfallu. Rigby, Sutherland a Takeuchi (2016a) se domnívají, že implementace agilních metodik může být poněkud problematická v takových oblastech, jako např. prodej, nákup, účetnictví nebo údržba.

Je přijatelné zavést agile pouze částečně: „*Agilně můžete řídit třeba jen jeden produkt a zbytek firmy může jet beze změny*“ (ŠOCHOVÁ, 2014, s. 120). Nebo ho implementovat postupně, třeba začít z nějakého určitého oddělení: „...*the most successful introductions of agile usually start small*“ (RIGBY, 2016a). Důležité je, aby agile nebyl slepým sledováním doporučení, která je možné najít ve člancích a učebnicích, ale aby šlo o hlubší prozkoumání všech aspektů agilních metod, bez kterého proces implementace může být komplikovanější. K zavedení Scrumu ve společnosti by měly vést určité důvody a měla by být provedena analýza současného stavu podniku. Kromě toho agilní metody jsou jen nástrojem. Jak poznamenává Myslín (2016, s. 29),

*„Žádná metodika, žádný princip ani žádná pravidla váš projekt nespasí, pakliže nebudete s nástroji umět zacházet“.*

Zde narážíme na skutečnost, že použití a prospěch implementace jakékoliv metodiky (nejen agilní) závisí ne na samotné metodice, ale na lidech, kteří s ní pracují: „...nejčastější příčinou selhání projektu je nikoliv selhání metodiky, ale obyčejné selhání lidského faktoru“ (MYSLÍN, 2016, s. 22). Stávají se situace, že agile implementují lidé, kteří mu neporozuměli tak, jak by bylo žádoucí, což vyvolává chybné představy o fungování projektů dle vybrané metodiky: „...because they haven't gone through training, they don't really understand the approach. Consequently, they unwittingly continue to manage in ways that run counter to agile principles and practices... <...> With the best of intentions, they erode the benefits that agile innovation can deliver.“ (RIGBY, 2016a). Na lidském faktoru závisí tedy všechno, co může být spojeno se řízením projektů. Na to poukazuje i jiná problematika, a to špatná komunikace týmu (nebo její nedostatek) se zbytkem organizace: dle výzkumu společnosti Scrum Alliance více než 70 % odborníků konstatují problém v komunikaci mezi týmy a jinými odděleními (RIGBY, 2016a).

Nicméně statistika nasvědčuje tomu, že agilní metodiky jsou více a více populární. O konkrétních číslech se pojednává dále.

#### 2.4.2 Agile a statistické údaje

Nejnámější výzkum, který se týká agilních metod, každoročně provádí společnost VersionOne. V roce 2017 byl publikován zatím poslední, 11. report, který poukázal na růst používání agilu napříč různými organizacemi a odvětvími – pouze 23 % dotazových pracuje ve společnostech spojených s vývojem softwaru (VersionOne, 2017, s. 2). Jinak jsou ve výzkumu zastoupené např. spořitelny, zdravotnictví, výroba, vzdělávání, letectví nebo neziskové organizace (VersionOne, 2017, s. 6).

Výzkumu se zúčastnilo přes 20 tisíc lidí z celého světa, nejvíc ze Severní Ameriky (50 %), z Evropy (28 %) a z Asie (10 %), (VersionOne, 2017, s. 5). Největší zastoupení měli projektoví manažeři (23 %) a Scrum Masteri (19 %), (VersionOne, 2017, s. 6).

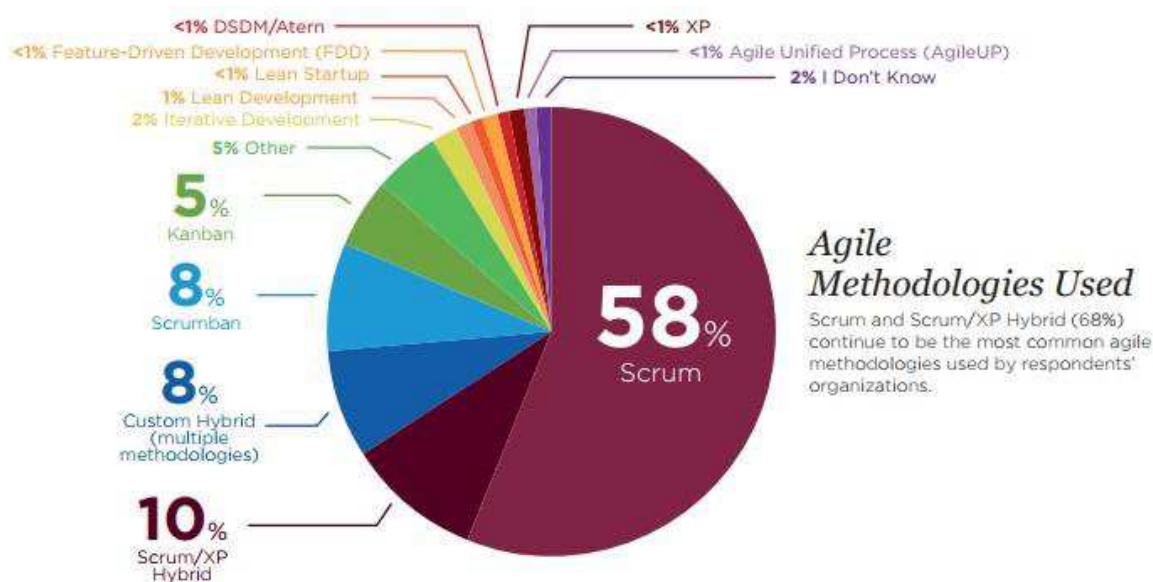
Vyčet některých výsledků k obecnému používání agilních metodik (VersionOne, 2017, s. 2):

- 98 % respondentů řeklo, že implementace agilu byla ve prospěch jejich organizace;
- 94 % respondent uvedlo, že jejich organizace agilní metodiky používá, ale zároveň 60 % odpovědělo, že to se týká méně než poloviny týmů v těchto společnostech;
- 80 % teprve s agilem začínají nebo jsou na střední úrovni ovládání těchto metodik;

- 71 % dotazovaných nyní investuje do DevOps.

Co se týká implementace agilu, tak k tomuto kroku vedly respondenty následující důvody: zrychlení procesu dodávání produktu (69 %), možnost řídit měnící se priority (61 %) a opět zvýšení produktivity týmu (53 %) <sup>13</sup>, (VersionOne, 2017, s. 8). Za největší přínosy agilních metod označují respondenti už šestý rok za sebou stejnou trojici, a to možnost řídit měnící se priority (88 %), zvýšení produktivity týmu (83 %) a lepší viditelnost projektu (83 %), (VersionOne, 2017, s. 9). Dá se říct, že po přechodu na agile získaly zkoumané organizace to, co si přáli.

Následující diagram demonstruje, jak jsou jednotlivé agilní metodiky a praktiky používány. Dle výsledků je vidět, že je na prvním místě Scrum, na druhém – hybridní metodika Scrum/XP, třetí místo si mezi sebou rozdělily jiné hybridy, a to Scrumban a „uživatelský hybrid“ – oboje po 8 % (VersionOne, 2017, s. 10).



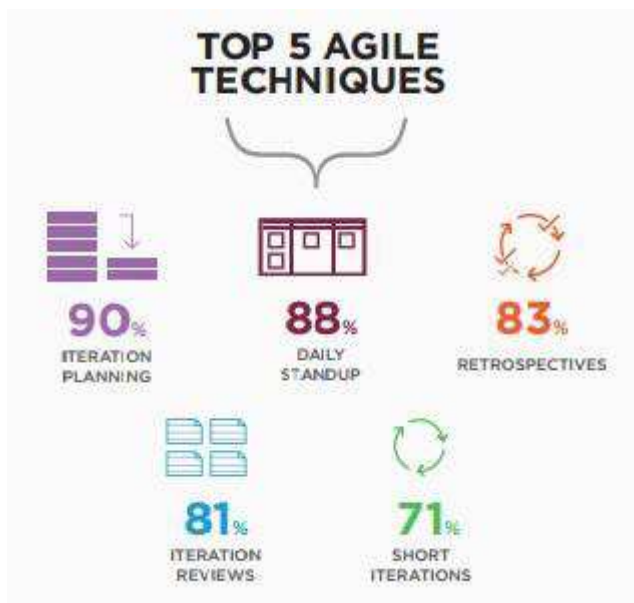
**Obrázek 1. Agilní metodiky a praktiky v roce 2017 dle VersionOne. Zdroj: VersionOne, 2017, s. 10**

Pokud jde o větší, komplexnější metodiky, vybrali respondenti jako nejlepší Scaled Agile Framework (SAFe) a Scrum (Scrum of Scrums), mající minimální rozdíl – jde o výsledky 28 % a 27 % (VersionOne, 2017, s. 14).

Druhý obrázek ukazuje, jaké agilní techniky získaly největší uznání mezi dotazovanými. Jak je vidět z procentuálních údajů, nejoblíbenější praktikou jsou iterace neboli sprinty; poslední

<sup>13</sup> U tohoto a následujícího dotazu mohli respondenti vybírat z několika možností.

termín je v českém prostředí obvykle preferovaný. Na druhém místě jsou daily stand-upy, na třetím – retrospektivy.



**Obrázek 2. Agilní techniky v roce 2017 dle VersionOne. Zdroj: VersionOne, 2017, s. 10**

Za největší překážky implementace a používání agilních metodik označují respondenti firemní filozofii a kulturu (63 %), nedostatek zkušeností se samotným agilem (47 %) a nedostatek podpory managementu (45 %), (VersionOne, 2017, s. 12).

Jsou k dispozici i údaje k různým nástrojům. Pokud jde o způsoby měření úspěšnosti, jako nejlepší byly zvoleny velocity (67 %), sprint burndown (51 %) a release burndown (38 %), (VersionOne, 2017, s. 13). Ohledně používaných programů se respondenti vyjádřili takto: Jira od společnosti Atlassian je používána u víc než poloviny dotazovaných (53 %), za ní následují Microsoft Excel (46 %) a Microsoft TFS<sup>14</sup> (22 %), (VersionOne, 2017, s. 16). Zajímavé je, že výsledky dotazu o doporučení programu jsou jiné. Tam první trojka vypadá takto: VersionOne (87 %), Jira (81 %) a CA Agile Central (76 %), (VersionOne, 2017, s. 17). V obou případech mohli respondenti vybírat z několika možností.

Společnost VersionOne nabízí zajímavé číselné údaje, získané z výsledků výzkumů, které mají kvantitativní povahu a probíhají každoročně. Další kapitola doplní tyto závěry analýzou a porovnáním několika výzkumů, které jsou z většiny kvalitativní a uvádějí poměrně různorodá data ohledně implementace a používání agilních metod.

<sup>14</sup> Team Foundation Server

### 3. Současný stav výzkumu v oblasti kritické reflexe agilních metod projektování

Cílem této části diplomové práce je zmapovat situaci kolem výzkumu agilních metod a ukázat, čemu se badatelé nejvíce věnují, jak zkoumají tento způsob řízení projektů, jaké jsou podněty vedoucí k implementaci agilních metod, jak identifikují a jaké doporučení pro zájemce o tento způsob řízení projektů mají. Důraz byl kladen na praktické zkušenosti lidí, jež používají agile. Nejlepší cestou k takové analýze je rešerše odpovídající literatury.

Tato konkrétní rešerše má omezení časové, jazykové a druhové. Byly akceptovány zdroje po roce 2010 až po rok 2018. Výsledky jsou převážně v angličtině, jako druhý jazyk byla k vyhledávání použita čeština. Druhově byla rešerše omezena na články v elektronických seriálech. Vysokoškolské závěrečné práce, stejně jako příspěvky z konferencí, nebyly brány v potaz. Teritoriálně soupis omezen není.

Byly využity různé rešeršní strategie: strategie rostoucí perly (postupné dodávání dalších klíčových slov) s využitím funkce „related articles“, pokud to vyhledávací systém umožňoval; strategie omezení/osekávání (použití filtrů) a sběr lesních plodů (hledání citací, procházení časopisů).

Určitý počet článků byl získán přes ScienceDirect, nebo tam odkazovaly DOI mnohých záznamů, které byly nalezeny přes EBSCOhost, Wiley Online Library nebo Springer. Také bylo provedeno vyhledávání přes Taylor & Francis Online, Emerald a ProQuest. Přístup byl zajištěn přes portál elektronických zdrojů Univerzity Karlovy nebo přes uživatelské rozhraní Národní technické knihovny. Discovery systém Ukaz.cuni.cz nabídl jen tři články v českém jazyce, jinak jsou všechny výsledky v angličtině.

Celkem bylo v angličtině nalezeno více než 150 článků k agilním metodám (s možností získat plný text a to z výše stanoveného časového období), avšak po průběžné analýze byly použité jen některé, neboť se vyskytovaly duplicity, dle tématu články neodpovídaly cílům výzkumu, nebo šlo pouze o sekundární či terciární analýzu literatury. Pro konečné rešerši bylo vybráno osm článků.

#### 3.1 České prostředí

Pokud jde o ČR, tak výzkumů není mnoho, přičemž každý z již existujících byl proveden před čtyřmi až pěti lety, což znamená, že informace z nalezených článků je poměrně stará a může

reflektovat dnešní situaci kolem používání agilních metod v ČR jen částečně. Avšak vzhledem k tomu, že jiné podobné prameny nebyly získány, budou v této diplomové práci oba dva výzkumy zmíněny.

Antlova (2014) zkoumá, jaké povědomí mají o agilních metodách pracovníci různých českých podniků. Provedla dotazování ve 40 společnostech s cílem dozvědět se, zda začaly být agilní metody používány více v porovnání s rokem 2007, kdy byl podobný výzkum uskutečněn jiným badatelem. Organizace měly různý profil: ICT, automobilový průmysl, výroba, služby a logistika (ANTLOVA, 2014, s. 932). Kontrolované rozhovory byly provedeny s projektovými managery. Zjistilo se, že nejvíc jsou tyto metody používány v ICT a automobilové výrobě, zatímco logistika a výroba toho mají méně, navíc služby agile vůbec nepoužívají (ANTLOVA, 2014, s. 932).

Za rok publikoval podobný výzkum Tománek (2015), který porovnává využití vybraných agilních metod (Scrumu a XP) v ČR a ve světě, a to na základě jak analýzy dvou výzkumů z roku 2013, tak i za pomoci výsledků jiného výzkumu ze stejného roku, který provedl sám autor. První zmíněný výzkum byl proveden společností VersionOne, jejíž statistické údaje byly již v této diplomové práci citovány. Druhý výzkum byl proveden Agilní Asociací<sup>15</sup> za pomoci společnosti Etnetera a byl změřen přímo na Českou republiku (TOMÁNEK, 2015, s. 5). Autor se zaměřil na porovnání dat ze dvou výše zmíněných výzkumů a svého průzkumu, který byl zaměřen na nejmenovanou mezinárodní logistickou společnost (TOMÁNEK, 2015, s. 11). Jde o průzkum ve formě dvoukolového dotazníkového šetření s cílem zjistit, „*které projektové týmy využívají agilní metodiky*“ (TOMÁNEK, 2015, s. 11). Počet vyplněných dotazníků se rovnal 452. Dle údajů, které Tománek (2015, s. 12) uvádí a porovnává, začal být agile v České republice široce používán později než ve světě, a to kolem roku 2010. Co se týká četnosti agilních projektů v ČR, tak 46 % respondentů (největší hodnota) uvedlo, že méně než 25 % projektů je řízeno agilní metodikou (TOMÁNEK, 2015, s. 13). Výsledek byl dán autorem na základě analýzy nejen jeho vlastního průzkumu, ale i výzkumu Agilní Asociace, a data logicky korelují s faktem pozdějšího zavádění agilu na území ČR. Autor doplnkově uvádí výčet nejrozšířenějších agilních technik, a to jsou „*denní schůzka*“<sup>16</sup>, „*plánování iterací*“<sup>17</sup>, „*retrospektiva a jednotkové testování*“ (TOMÁNEK, 2015, s. 14). Zajímavým postřehem je zmínka o tom, že jsou v ČR častěji než ve světě využívány agilní metodiky se zaměřením na technické aspekty

---

<sup>15</sup> Agilní Asociace, „*sdržení příznivců agilních metod*“ (AGILNÍ ASOCIACE, 2018), byla založena Šochovou, Kuncem a Šukem v roce 2011. Provádí nejrůznější akce týkající se agilních metodik a mezi nejznámější patří konference Agile Prague (AGILNÍ ASOCIACE, 2018).

<sup>16</sup> V diplomové práci je používán anglický termín „daily stand-up“.

<sup>17</sup> Jde o plánování sprintů.

vývoje softwaru (TOMÁNEK, 2015, s. 14). Jako hlavní důvod, proč agile nebyl zaveden, se v ČR ukazuje „vědomostní deficit“, k největším obavám patří „*strach ze ztráty počátečního plánování a ztráty manažerské kontroly nad vývojem softwaru*“ (TOMÁNEK, 2015, s. 14). Na konec analýzy by bylo vhodné uvést citaci ze závěru článku: „*Postupné nasazování agilních metodik a jejich zvyšující se obliba přispívá k celkovému pozitivnímu vnímání agilních metodik. Do budoucna proto používání agilních metodik bude dle autora dále růst.*“ (TOMÁNEK, 2015, s. 15).

Dá se říct, že tyto dva výzkumy se navzájem doplňují, avšak z dnešního hlediska jsou poměrně staré pro analýzu situace v České republice. Nicméně je z nich možné dedukovat, že před čtyřmi až pěti lety byl agile používán docela málo a z většiny šlo o společnosti spojené s ICT sektorem. Oba autoři použili metodu dotazování, Antlova (2014) doplnkově provedla osobní rozhovory.

Dalších šest výzkumů je představeno v následující části této kapitoly.

### 3.2 Zahraniční výzkumy

Vzhledem k tomu, že jsou výzkumy poměrně různorodé jak dle cílů zkoumání, tak i dle času publikování, rozhodla jsem se provést analýzu každého z nich zvlášť, používajíc některé společné aspekty jako logické spojky. Tato část třetí kapitoly se začíná články zkoumajícími to, jak a proč se agile implementuje, a končí pracemi, které analyzují následky tohoto procesu pro společnost. Tím pádem je důraz kladen na obsah, čas publikování nemá vliv na pořadí. Všechny výzkumy lze charakterizovat jako případové studie (case study), což nejlépe odpovídá záměrům diplomové práce.

První analyzovaný výzkum je od Trippa a Armstronga (2016), kteří se ve svém výzkumu zaměřují na to, jaké faktory povzbuzují společnosti k přechodu na agile a jak vybrané jimi agilní techniky ovlivňuje proces implementace a její výsledky neboli její úspěšnost. Jako zdroj dat používají jeden z výzkumů VersionOne (stejně jak to udělal i Tománek (2015) o něco dříve) a své polostrukturované rozhovory se specialisty, kteří agilní metody aplikují. Respondenti mají v průměru 27 let zkušeností v oblasti softwarového vývoje a jsou buď agilní kouči, nebo mají ve vývoji vedoucí pozici (TRIPP, 2016, s. 171). Motivační faktory byly rozděleny do kategorií. První kategorie se týká zlepšení kvality softwaru a skládá se z takových motivů jako zvýšení kvality produktu, zdokonalení disciplíny a zlepšení udržitelnosti softwaru (TRIPP, 2016, s. 172). Druhá kategorie je věnovaná zlepšení efektivity (v originálu *Efficiency*) a zahrnuje zvýšení produktivity, zrychlení času potřebného na dodání produktu na trh a snížení nákladů

(TRIPP, 2016, s. 172). Třetí kategorie, zlepšení efektivnosti (*Effectiveness*), se soustředila na takové motivy k implementaci agilu, jako schopnost organizace řídit měnící se priority a zlepšení kooperace mezi businesssem a IT (TRIPP, 2016, s. 172). Co se týká agilních technik, těch bylo vybráno 12 a byly rozděleny na takové kategorie, jako projektový management a přístup k vývoji softwaru.

Autoři provedli analýzu vzájemného poměru mezi motivačními faktory organizací a vybranými agilními technikami. Bylo zjištěno, že pokud je cílem společnosti zvýšení kvality softwaru, je pozitivní korelace mezi motivem a kapacitou zvolených technik z kategorie „přístup k vývoji softwaru“ a negativní korelace s druhou kategorií. Pokud jde o zlepšení efektivity, je zde pozitivní korelace s kapacitou zvolených technik z kategorie „projektový management“ a nulová korelace s jinou kategorií agilních technik. V případě zlepšení efektivnosti je pozitivní korelace jak s vývojem softwaru, tak i s projektovým managementem. Čím více příslušných technik společnost implementuje v souladu s motivačními faktory, tím bude lepší výsledek. (TRIPP, 2016, s. 174)

Autoři doporučují organizacím, které jsou zaměřené na zvýšení kvality produktu nebo na zlepšení efektivnosti, implementovat více technik k vývoji softwaru, zatímco pokud jde o zvýšení efektivity, je nutné se soustředit na techniky projektového managementu (TRIPP, 2016, s. 176).

Na konci autoři uvádí doporučení k implementaci agilních metod na základě proběhlých rozhovorů, z toho tři pro společnost a čtvrté pro tým (TRIPP, 2016, s. 176–177):

- Na začátku se soustředit na nejproblematictější místa a odtud metodiky rozšiřovat – provádět implementaci postupně;
- Nastavit klíčové ukazatele úspěšnosti (*key performance metrics*);
- Zpřesnit motivy k implementaci, metriky a agilní praktiky (důležité pro pochopení zaměstnanci, proč se to vůbec děje);
- Týmy by měly implementovat několik agilních praktik, ale vybrat je s rozumem.

Následné doporučení k implementaci agilu dala pro případné zájemce i skupina autorů (PIKKARAINEN, 2012), kteří publikovali výsledky výzkumu, kterým se pokusili zjistit, jaké silné stránky a zároveň bariéry mají agilní metody. Odlišností od předchozích autorů bylo to, že zde nebyly zkoumány faktory ovlivňující implementaci, ale šlo už o to, co negativního a pozitivního tyto metodiky přinesly pro společnost. Pro výzkum, který trval 2,5 roku mezi léty 2005 a 2008, byly záměrně vybrány pilotní projekty, a to ve třech finských společnostech, které



jsou zaměřené na vývoj softwaru a chtějí zavést agile. První společnost má finanční zaměření a kolem osmi tisíc pracovníků, druhá – se zaměřením na telekomunikace, má kolem sta lidí, třetí – na informační bezpečnost, má kolem tří set zaměstnanců (PIKKARAINEN, 2012, s. 682). Případová studie zahrnovala pozorování, 12 seminářů a 40 polostrukturovaných rozhovorů – individuální a skupinové (PIKKARAINEN, 2012, s. 680). Mezi 57 respondentů patřili vývojáři, architekti a různí manažeři, kteří si navíc po iteracích a seminářích zapisovali silné stránky a bariéry, které byly poté posuzovány skupinově kvůli zvýšení objektivního přístupu.

Ze třech společností dvě řekly, že projekt byl úspěšný a že použití agilních technik bylo užitečné. První společnost spokojená nebyla, ale je zajímavé, že i když cílem bylo implementovat Scrum, tak se tam se používaly pouze daily stand-upy a párové programování (PIKKARAINEN, 2012, s. 683–684). Ovšem i přes zmíněné hodnotil první tým docela pozitivně rozšíření spolupráce s jinými odděleními a aktivnější účast stakeholderů. Druhá skupina implementovala některé praktiky Scrumu, XP, a Mobile-D™, a jejich výčet je delší v porovnání s prvním týmem. Respondenti byli spokojení a označovali, že jejich produktivita se zvýšila o 20 % (PIKKARAINEN, 2012, s. 685). Navíc se zlepšila komunikace a pochopení potřeb zákazníka. Lidé z třetí společnosti ocenili využití agilních praktik jako velice užitečné. Implementovali prvky Scrumu, XP and Mobile-D™, stejně jako druhá společnost. Označili v rozhovorech, že vývoj byl třikrát rychlejší, padesátkrát kvalitnější a pětkrát levnější (PIKKARAINEN, 2012, s. 685).

Co se týká silných stránek a bariér, byla také každá skupina posuzována zvlášť. Za největší problémy v první skupině byly označeny: nedostatek jasnosti ohledně nasazení agilu ze strany vyššího managementu, absence náležité přípravy u týmu (implementovali to, co za agilní považovali) a absence potřebných znalostí u vývojářů (PIKKARAINEN, 2012, s. 688). Zároveň se nevyskytovaly žádné systematické kontroly zlepšení, pokroku týmové práce. Ve druhém případě, kdy chtěla společnost zkusit agile poté, co zaslechla o úspěšnosti daných metodik, byly už před začátkem pokusu o implementaci někteří pracovníky proti agilu, což ovlivnilo průběh práce; po seminářích stejně ne všichni chápali smysl agilních metod, kvůli čemuž bylo těžké agilní techniky uplatňovat, ale to se po čase zlepšilo (PIKKARAINEN, 2012, s. 690). Vliv mělo i to, že lidé poměrně často pracovali napříč různými projekty, nebo že zákazník chtěl koupit už hotový produkt, což se také pravidelně stávalo (PIKKARAINEN, 2012, s. 691). Na odlišení od předchozích případů, ve třetí společnosti před zahájením projektů udělali dvoutýdenní školení k agilním metodám, a agilní techniky byly uplatněny nejen na vývoj softwaru, ale i v marketingu a řízení. Také je nutné uvést, že třetí společnost chtěla zkusit agilní přístup kvůli

potřebnosti reagovat na dynamickou, měnící se situaci na trhu (PIKKARAINEN, 2012, s. 692). Tým zde, jako i v prvním případě, nedělal systematickou kontrolu a měl problémy se zlepšením procesu. Problémy se vyskytovaly ve spolupráci se stakeholdery, kteří nevěděli, jak agile funguje a jaké role mají. Negativní vliv měla změna Scrum Mastera a komunikace s týmy, které pracovaly dle neagilních metod. Celková analýza ukazuje, že nejsilnější stránky agilu jsou komunikace uvnitř týmu a se stakeholdery (jde o obecné rysy všech třech společností), (PIKKARAINEN, 2012, s. 693). Bariéry se lišily tým od týmu a jsou rozepsány výš, stejně jako příčiny těchto problémů. Autoři podtrhují, že změny jsou potřebné nejen na úrovni týmu, ale i v celé organizaci, a že implementace potřebuje patřičné znalosti a pochopení toho, co se děje, i ze strany vedení firmy, a to např. v její strategii (PIKKARAINEN, 2012, s. 694). Co se týká dokumentace, její velikost a rozsah záleží na kontextu a potřebách situace. Jako finální doporučení autoři definují (PIKKARAINEN, 2012, s. 694–695):

- Podpora managementu;
- Vize, pochopení a uvědomění si toho, co agilní metodiky představují u každého, kdo k tomu má co do činění;
- Tým potřebuje určitou svobodu pro náležitou implementaci agilních metod;
- Nepřetržitou úpravu metodik je nutné provádět i na organizační úrovni.

Příklad toho, jak je používán agile ve velké firmě, je popsán i v jiném článku, který je zaměřen na zkoumání požadavků, jejich řízení v agilní společnosti (HEIKKILÄ, 2017). Autoři provedli výzkum ve společnosti Ericsson. Zajímalo je, jak jsou řízeny požadavky od stanovení strategie do vydání produktu, jaké role a procesy jsou s tím spojené, jaké výhody a problémy mohou být předpokládány (tím se podobají předchozímu výzkumu). Upozorňují, že implementace a adaptace agilních metod v natolik velkých společnostech se liší od menších, a poukazují, že v takových případech jsou obvykle implementovány buď Large-Scale Scrum (LeSS) nebo Scaled Agile Framework (SAFe) jako nejpopulárnější v tomto segmentu (HEIKKILÄ, 2017, s. 2897). Data byla sebrána na základě rozhovorů a analyzována kvalitativně. Prvního kola se zúčastnilo osm specialistů, druhého 19 z Finska a 11 z Maďarska. Obě kola měla za cíl pochopit organizaci a procesy. Třetí kolo bylo zaměřeno na řízení požadavků a zúčastnilo se ho pět lidí. Je nutné zmínit, že na začátku fungovala společnost dle waterfallu, ale pak vedení vybralo Scrum jako nejvhodnější metodiku pro současné potřeby (HEIKKILÄ, 2017, s. 2902–2903). Zajímavostí je, že kvůli velikosti podniku není u produktu jediný Product Owner, ale existuje celá skupina, která zahrnuje Chief PO, ředitele tohoto týmu, a deset PO, kteří mají pro minimalizaci rizik stejnou odpovědnost za požadavky, řízení a synchronizaci práce týmů (HEIKKILÄ, 2017, s. 2906). Scrum byl implementován jen částečně.

Jak už bylo řečeno výše, autoři kromě jiného zkoumali benefity a problémy agilního způsobu vedení projektů, konkrétněji správu požadavků, což je podle Scrumu to nejpodstatnější při jednání se zákazníkem. K benefitům patří dle výzkumu zkrácená doba vývoje, vyšší flexibilita, zlepšení efektivnosti procesu plánování, zvýšená motivace vývojářů, přímé a efektivní spojení všech dotyčných (HEIKKILÄ, 2017, s. 2913–2915). Co se týká problémů, jedná se zejména o: tlak ze strany product managementa (který, dle respondentů, pracoval starým způsobem), problematické plánování požadavků (prosba o detailech ze strany product managementu versus přání organizace zachovat nižší detailizaci pro lepší flexibilitu), plánování na úrovni systému, dokumentace a řešení problémů (kdo to má řešit), definování odpovědnosti PO (členové PO týmů měli velice různorodé zkušenosti, kvůli čemuž bylo obtížné řešit např. technické záležitosti), růst tzv. „technického dluhu“ (přednost byla daná vyplňování user stories, při tom trpěla vnitřní kvalita systému), vyrovnání mezi efektivitou vývoje a cross-funkční týmy, nedostatečné pochopení nezávislosti týmu (vývojáři čekali, že jim PO nebo SM řekne, co je nutné dělat; s postupem času se problém zmenšil), (HEIKKILÄ, 2017, s. 2915–2918).

Dle mínění autorů je k řízení požadavků možné použít hierarchický model, což může být obzvlášť užitečné pro velké společnosti, ale je zároveň nutné klást důraz i na jiné, příbuzné, oblasti (HEIKKILÄ, 2017, s. 2919–2920). Užitečné by mohlo být i rozdělení týmů dle technických oblastí pro snížení nákladů. Pokud jsou problémy s týmovou komunikací a koordinací, autoři doporučují vytvořit ještě jeden tým se zaměřením přímo na koordinaci, který by byl v hierarchické pozici výš než v daném případě tým PO; zároveň označují, že metodika Scrum of Scrums pro řešení tohoto problému není dostačující (HEIKKILÄ, 2017, s. 2925). Pro eliminaci technického dluhu může být pomocná role architekta nebo technického specialisty. Jeden z výsledků, který autoři získali, praví, že agilní metody lze efektivně spojit i s waterfallem v rámci organizace: *„Subsequently, it is not necessary to completely disassemble and reorganize the whole organization in order to gain the benefits of agile development on the team level.“* (HEIKKILÄ, 2017, s. 2926).

Heikkilä (2017) zmínil kromě jiného i takovou problematickou stránku implementace agilních metod, jako nastavení týmové komunikace a spolupráce všech členů týmu. Pokud v tomto výzkumu šlo o spojení několika týmů ve velkém podniku, Henriksen a Pedersen (2017) zkoumali, jak mohou agilní metody ovlivnit úspěch projektu při práci malých týmů. To se uskutečnilo za pomoci rozhovorů se čtyřmi respondenty ze dvou odlišných norských projektů a doplňkového hodnocení předem vybraných praktik. V každém týmu byl jeden člověk Product Owner a druhý – vývojář, pracující na Sri Lance. Týmy používaly převážně Scrum. Za pomoci

literární rešerše bylo identifikováno 53 používajících se praktik (např. pouze k XP bylo nalezeno 24 praktik), 15 z toho bylo uznáno jako obzvlášť důležité pro agilní vývoj softwaru, i když mezi týmy byly některé odlišnosti ohledně hodnocení (HENRIKSEN, 2017, s. 71). Šest praktik se týkalo kvality produktu, osm bylo spojeno s rozsahem prací, a jedna – s časem. Jako nejdůležitější byly vnímány takové praktiky, jako iterativní řízení a sprint review (kvůli rannímu feedbacku od zákazníka), tzv. „incremental design“ (kvůli měnícím se požadavkům a prioritám zákazníka) a sprint retrospective (pro celkové zlepšení týmové práce), (HENRIKSEN, 2017, s. 71). Jiných 10 praktik bylo vysoce hodnoceno díky růstu transparentnosti, lepšímu plánování, zlepšení přehledu projektu, rozvoji komunikace v týmu a zlepšení řízení času, který byl na vývoj produktu potřebný (HENRIKSEN, 2017, s. 71). Jinak měla na lepší vývoj produktu vliv kvalita a rozsah, čas tak důležitý nebyl. Klíčové jsou odezva zákazníka, pochopení jeho potřeb a týmová práce. Výsledky ukázaly, že přístup, který pomáhá pochopit potřeby zákazníka, zvyšuje jeho spokojenost a zlepšuje týmovou práci, bude mít vliv na úspěšnost projektu.

Pokud byl daný výzkum zaměřen spíše na to, jak nastavit, popřípadě zlepšit týmovou komunikaci, Moe, Dingsøyr a Dybå (2010) se zaměřili na to, proč se objevují problémy v týmové práci při implementaci takového agilního prvku jako samoorganizovaný tým. Autoři se snažili pochopit, jak funguje agilní tým, jaké má složení, protože, dle jejich názorů, to může pomoci v budoucnu zlepšit a zefektivnit celý proces implementace agilních metod (MOE, 2010, s. 480). Používajíc Dickinson a McIntyre's teamwork model, Moe, Dingsøyr a Dybå (2010, s. 483) vytvořili základ pro svůj výzkum, který klade důraz na lidské emoce při společné práci a na pochopení tohoto procesu. Samotný výzkum, který probíhal v terénu jedné společnosti, přesněji v odděleném ICT oddělení, měl tři fáze a zahrnoval pozorování, rozhovory a besedy se členy týmu o projektu, jeho pokroku a o tom, jaké problémy se objevily (MOE, 2010, s. 483). Výzkumu se zúčastnili i lidé s jinými rolmi. Důležité je, že projekt, který autoři pozorovali, byl ve společnosti první s použitím Scrumu, ale jednalo se o Scrum s použitím tzv. „izomorfní týmové struktury“ – každý odpovídal za svůj modul od začátku do konce, což výrazně ovlivnilo práci nad projektem (MOE, 2010, s. 485). Nad ním pracovali Product Owner, Scrum Master a šestičlenný tým a pro většinu to byla první podobná zkušenost. Celkem bylo zdokumentováno 60 pozorování (což byly i daily stand-upy, sprint review a jiné meetingy) a neoficiálních besed; navíc k tomu bylo provedeno 17 polostrukturovaných rozhovorů, a to jak na začátku projektů, tak i po jeho ukončení – celkem devět měsíců (MOE, 2010, s. 483, s. 489). Autoři pozorovali problémy s vyplněním backlogu, neúčinné meetingy, mlčení členů týmu na plánovacích meetingách nebo práci s jinými úkoly, než které byly předtím nahlášené (MOE, 2010, s. 485).

Na začátku (první dva sprinty; každý sprint v projektu, kromě posledního trval měsíc) lidé často dělali to, co uměli, ale ne to, co bylo pro tento sprint potřebné, tím dávali přednost individuálním přáním, ale ne cílům týmu, a při volbě úkolu a jeho vyplňování ho vnímali jako personální záležitost (MOE, 2010, s. 486). Takový přístup vedl k separaci některých lidí z týmu a jejich menší spokojenosti – komunikace se zhoršila, někdo vůbec neposlouchal ostatní, protože ho to nezajímalo; feedback byl minimální, monitorování skoro žádné, chyběly rezervní kopie (MOE, 2010, s. 486). Takové byly problémy i s adaptací k měnícím se požadavkům zákazníka. Kromě toho nebylo o mnohých problémech řečeno Scrum Masteru, protože se členové týmu domnívali, že reaguje příliš ostře. Za nějakou dobu (3–4 sprinty) se tým navíc ztratil, protože mu nebyla poskytnuta podpora od vyššího vedení, ale o něco se zlepšila komunikace. Dle členů týmu byly daily stand-upy lepší bez Scrum Mastera, protože při jeho absenci mohou mluvit jeden s druhým (MOE, 2010, s. 486). Vyskytovaly se problémy i ze strany samotného Scrum Mastera, který spíše kontroloval a přikazoval, než aby podporoval a řídil. Chyběla vzájemná důvěra, a to jak mezi členy týmu, tak i mezi nimi a Scrum Masterem. Lze říct, že problémy týmu byly už od samého začátku nebo i před tím, kvůli čemuž tým nedosáhl stavu samoorganizace.

Možnost, jak podobnou situaci zlepšit, může být spojení Scrumu s XP. Použití techniky párového programování by mohlo pomoci s monitorováním, feedbackem a rezervními kopiemi. Také není doporučován izomorfní model, jak to bylo v tomto případě. Problémem může být i vysoká individuální autonomie. Je nutná i podpora nových praktik, omezení starých, více času a lidí. Byla by zapotřebí i místa pro vzdělávání nebo přemýšlení. Pro přechod na agilní metody je potřebný čas a podpora, a to nejen vzájemná v týmu, ale i od vyššího managementu. (MOE, 2010, s. 489)

Co se děje ve společnosti poté, co zde byly agilní metody už vyzkoušené, se snaží zjistit poslední zde popsany výzkum, který provedli Senapathi a Srinivasan (2012) na Novém Zélandu. Zajímala je identifikace faktorů ovlivňující použití a efektivitu agilních metod, a to, jak intenzivně je agile ve společnosti používán (ne samotný proces implementace jako takový), přičemž důraz je kladen ne na nějakou jednu metodiku, např. Scrum, ale na soubor technik, které z pohledu společnosti vyhovují jejím cílům a vizím co nejlépe, a díky tomu je jako kombinace prvků implementován (SENAPATHI, 2012, s. 1258). Předpokládá se, že je vztah mezi stupněm využití (horizontální – různé projekty, týmy atd., vertikální – maximální využití konkrétních praktik) a efektivitou využití agilu (produktivita, kvalita a spokojenost zákazníka), (SENAPATHI, 2012, s. 1259).

Autoři nabídli integrační model, který identifikuje deset klíčových faktorů pro budoucí využití agilu. První dva faktory jsou inovační – relativní výhoda (v porovnání s předchozími metodikami) a slučitelnost (s již existujícími zkušenostmi a kontextem společnosti). Sociologické faktory zahrnují zkušenosti týmu, přístup (negativní či pozitivní k agilu) a technické dovednosti. Technologické faktory jsou o agilních praktikách a podpoře nástrojů (např. speciálně určených pro agilní metodiky). Organizační faktory se týkají podpory top managementu a metodologii. Každý faktor byl vybrán a ověřen za pomoci literární rešerše. (SENAPATHI, 2012, s. 1258–1259)

Model byl ověřen ve dvou společnostech. První používala Kanban po zkušenostech se Scrumem, druhá – Scrum po waterfallu, a to trvalo více než dva roky. Bylo provedeno 17 polostrukturovaných rozhovorů s vývojáři, Scrum Mastery, business analytiky atd. Pokud první společnost neměla v podstatě žádné potíže, a to díky vysoké motivaci a předchozím zkušenostem se Scrumem, druhá po přechodu z waterfallu čelila většímu spektru problémů. Postoj k novému způsobu řízení projektů byl smíšený; největší odpor měli business analytici, ale i celkově byly poměrně hodně používány waterfallové metodiky a principy kvůli nedostatku znalosti Scrumu a zvyku pracovat ve starých rolích (SENAPATHI, 2012, s. 1265–1266). Respondenti v druhém týmu označili spokojenost a zapojení ze strany zákazníka za velice důležitý bod; navíc hrála v obou případech dle respondentů klíčovou roli podpora managementu; dá se říct, že to koreluje i s předchozím výzkumem (SENAPATHI, 2012, s. 1265). Avšak v tom, co se týká posledního zmíněného bodu, v obojích případech top management chtěl na začátku dobré výsledky před tím, než týmy mohli i nadále pokračovat s agilními metodami. Ale i později byl cítit nedostatek této podpory, protože vedení nechápalo, jak se proces změnil, a že agilní metody potřebují neustálá zlepšení, a feedback je proto velice důležitý (SENAPATHI, 2012, s. 1265). Celkové výsledky ukázaly, že všechny výše uvedené faktory mají vliv na efektivní využití agilních metod. Obzvlášť důležité je mít vysokou úroveň technických znalostí nebo ji rozvíjet za pomoci školení a podpory top managementu, který je obecně považován za klíčový pro úspěšné adoptování agilních metod nejen jedním týmem, ale i celou společností (SENAPATHI, 2012, s. 1267). Doplnkově respondenti poukázali na důležitost takových mimomodelových faktorů, jako čas a rozpočet, což dokazuje to, že model je nutné rozšiřovat a aplikovat na větším vzorci k získání doplňkových dat.

K tomu, co se týká dat a doplňkových bádání, je nutné říct, že v některých člancích (HENRIKSEN, 2017, s. 63; HEIKKILÄ, 2017, s. 2896) upozorňovali autoři na nedostatek kvalitních výzkumů ve zkoumání agilních metod řízení projektů; chybí i kvantitativní

výzkumy. Moe, Dingsøyr a Dybå (2010, s. 481) poukázali na to, že našli pouze jedinou výzkumnou práci o Scrumu do roku 2006.

Výše popsané studie a závěry posloužily jako inspirace pro vytvoření většiny otázek pro výzkum, který je základem praktické části této diplomové práce. Výsledky výzkumů ovlivnily i následná doporučení, které jsou uvedené v závěru práce.

## 4. Praktická část – kvalitativní výzkum v konkrétních společnostech

Jak bylo zmíněno už v úvodu této diplomové práce, praktická část je zaměřena na mapování stávající situace týkající se implementace a použití agilních metod ve vybraných českých společnostech, a to za pomoci kvalitativního výzkumu.

Na začátku jsou představeny metodologie výzkumu a jeho popis, dále budou provedené rozhovory analyzovány. Ve výstupu budou předložena doporučení pro implementaci agilních metod a dodržování filozofie agilu, přičemž tato doporučení budou sestavena jak na základě výzkumných výstupů, tak i na základě odpovídající literatury (viz předchozí kapitolu).

### 4.1 Metodologie a popis výzkumu

Výzkum byl proveden za pomoci sjednocení dvou metod: analýzy literatury k danému tématu a polostrukturovaných rozhovorů s respondenty.

Jelikož je celý výzkum zaměřen na analýzu procesu implementací agilních metod v českých podnicích a doplňkově je zkoumáno, jak jsou principy agilu dodržovány v realitě, byly proto výzkumné otázky pro respondenty rozdělené do několika logických úseků: potenciální přínos agilu, problémy při implementaci, principy a filozofie agilu, hodnocení výsledků implementace, následné doporučení. Tyto okruhy otázek byly mimo jiné sestaveny i na základě proběhlé literární rešerše, která byla spolu s prvním rozhovorem zpracována v rámci předmětu „Výzkumné metody“ (kód AIS510291) na Ústavu informačních studií a knihovnictví pod vedením paní doktorky Slussareff za pomoci dvou kolegů.

Některé výzkumné otázky (podrobněji viz Příloha 1):

- Proč je dnes důležité se zabývat agilními metodami?
- Jaké výhody a nevýhody mají podle Vašeho názoru agilní metody v porovnání s waterfallem?
- Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily?
- Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.



- Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?

Otázek je celkem 28, každá je v českém jazyce. Jsou koncipovány jako otevřené, ale některé z nich mají doplňkové podotázky, jako např. poslední otázka z předložených výše. Každá je sestrojena tak, aby neobsahovala dvojitý zápor, nebyla sugestivní a tím mohla přispět ke snížení pravděpodobnosti získání zkreslené informace. Po prvotním návrhu otázek a jejich kategorizaci nebylo nic změněno. Ukázalo se, že každá otázka je srozumitelná a respondenti jsou schopni odpovědět na všechny dotazy.

Před rozhovory, které pobíhaly při osobním setkání, byl každý respondent seznámen s otázkami prostřednictvím e-mailu. Před zahájením samotné besedy byl pokaždé získán souhlas respondenta s nahráváním rozhovoru na diktafon. Rozhovory se uskutečňovaly v češtině. Přepisy nahrávek byly zpětně odeslány na schválení pro zajištění vyšší kvality analýzy a interpretace výsledků.

Celkový počet respondentů je tři. V této diplomové práci jsou pro ochranu jejich anonymity použita označení R1, R2 a R3 (v samotných nahrávkách se také nevyskytují osobní jména).

R1 pracuje jako enterprise architekt v jedné společnosti v bankovním sektoru (XY). Osobní zkušenost s agilem má přibližně rok (Scrum). Rozhovor trval 45 minut a jeho přepis je možné nalézt v Příloze 2. Doplněk k rozhovoru byl poslán e-mailem a je součástí Přílohy 5.

R2 představuje logistický sektor (NN IT Services), pracuje jako software engineering director. Osobní zkušenost s agilem má kolem 30 let, a to s různými metodikami. Rozhovor trval přibližně 2,5 hodiny a zahrnuje komentáře nebo odchylky, které nepatří přímo k otázkám rozhovoru, kvůli čemuž bylo nutné částečně zkrátit přepis. Stejně tak zkrácená verze byla následně poslána respondentovi na schválení. Text lze najít v Příloze 3.

R3 je programátor pracující jak na sebe (projekty na zakázku), tak i v různých startupech, naposledy pracoval v XXX. Má zkušenosti i z větších firem. Dělal i Scrum Mastera. V práci byly vždy používány agilní metodiky, většinou Scrum. Zkušenosti s agilem má pět let. Rozhovor trval hodinu a jeho přepis je součástí Přílohy 4.

## 4.2 Analýzy rozhovorů

Jak bylo zmíněno dříve, otázky k rozhovorům byly rozděleny do několika okruhů pro lepší orientaci respondentů a usnadnění následné analýzy, která tvoří podstatu této podkapitoly. Struktura okruhů je zachovaná v prvotní podobě. Ke každé otázce jsou uvedeny odpovědi všech

respondentů pro názorné porovnání různých mínění ke stejné problematice. Pokud respondent řekl něco užitečného k jedné otázce v odpovědi na jinou, je ten určitý komentář uveden v příslušném místě dle obsahu, tj. ve druhé otázce pro menší rozptýl. Po analýze všech okruhů budou následovat všeobecná doporučení. Po každém okruhu je dáno stručné shrnutí.

Součástí výzkumu je i frekvenční analýzy textu, která bude uvedena v poslední části dané kapitoly.

#### 4.2.1 Okruh 1: Obecné (úvod)

##### 1. Připadají Vám agilní metody přínosné, nebo ne?

V odpovědi na tuto otázku byli respondenti jednomyslní – ano, agilní metodiky přínosné jsou. Avšak pokud R2 a R3 použili taková slova, jako „rozhodně“ a „samozřejmě“, R1 byl ve svém úsudku opatrnější. Řekl, že *„záleží na tom správném použití, takže podle mého ta správnost toho nasazení té metodiky dané je to, co potom z toho dělá tu přínosnou metodu“* (Příloha 2). Míní tím, že agilní metodiky mají své výhody a nevýhody, a je nutné to *„zacílit správným způsobem“* (Příloha 2), aby společnost mohla těžit právě ty výhody. Označil, že se příslušné procesy v XY podařilo nastavit správně, proto z toho mají užitek, kvůli čemuž je pro ně používání agilních metodik v tuto chvíli přínosné.

##### 2. Proč je dnes důležité se zabývat agilními metodami?

Jako u předchozí otázky, R2 a R3 se jednoznačně vyjádřili k tomu, že ano, dnes existuje potřeba agilních metod; dle R3, *„měly by se tím firmy zabývat“* (Příloha 4). R2 upozorňuje na důležitost agilních metodik z pohledu konkurenceschopnosti: *„Já si myslím, že pro společnosti, které vyvíjí softwarový produkt, tak je to dneska základní metoda vývoje, kterou když neadoptují, tak se budou těžko přizpůsobovat tomu trhu nebo těm měnícím se požadavkům na tom trhu. Podle mě, je to otázka přežití.“* (Příloha 3). R1 o tom uvažuje podobně – už v odpovědi k první otázce poukázal na to, že *„banky obecně mají problém v tuhlentu chvíli“* (Příloha 2), a to kvůli konkurenci ze strany malých firem neboli startupů, které jsou rychlejší, flexibilnější, což nutí větší podniky zabývat se agilními metodami kvůli změně poskytovaných produktů a fungujících procesů. R3 ještě dodává, že implementovat agilní metody se dá kdekoliv, nejen ve vývoji softwaru.

##### 3. Jaké výhody a nevýhody mají podle Vašeho názoru agilní metody v porovnání s waterfallem?

R1 za hlavní nevýhodu agilních metodik označuje to, že přístup potřebuje seniorní lidi, kteří mají náležité znalosti a schopnosti pro práci v agilních týmech, pro realizaci změn adaptivním způsobem, jak to teď v XY probíhá a kde se právě řešila kompletní strukturální změna ohledně agilu: „...já tam vidím různá velká rizika. Že nebudeme schopní najít lidi, kteří by fungovali, a vlastně juniorita těch lidí zruinuje výhodu těch agilních metod.“ (Příloha 2). Navíc poukazuje na to, že pokud ve společnosti pracuje více agilních týmů, tak Scrum neřeší spolupráci, koordinaci mezi nimi. Dokonce, dle názoru respondenta, může dojít k určité autonomii, což může ve výsledku způsobit i jakousi anarchii. Zmínil, že na rozdíl od Scrumu řeší problém Scaled Agile for Enterprise (SAFe) nebo LeSS (Příloha 2).

Zajímavým postřehem ze strany R2 je to, že „...waterfall byl vymyšlen Scrumem. Prostě historický tomu nikdo „waterfall“ neřikal. Prostě to byly nějaké metodiky softwarového vývoje, ale teprve když se vytvořil Scrum, tak vůči tomu se začali ukazovat – to je waterfall.“ (Příloha 3). Dle názoru R2, největší přínos agilních metod je v tom, že zákazník získá právě to, co si přeje, a to i za pomoci reakce na změnu požadavků, což waterfall nemá, protože spravuje spoustu obecných požadavků, které se pak nemění a možná nejsou pro zákazníka ani potřebné. Také označuje, že agilní metodiky, popřípadě konkrétně Scrum, jde snadněji implementovat: „Na rozdíl třeba od waterfallu nebo nějakých tradičních metod, které jsou velmi složité a z hlediska toho procesu jsou těžkopádné a těžko se implementují, tak ten Scrum podle mě je možné nasadit celkem rychle. Myslím si, že je to o rychlosti vývoje a o kvalitě.“ (Příloha 3). Zároveň však vyjadřuje mínění, že v některých případech je waterfall jasnější, lepší, efektivnější, a jako příklad uvádí situaci, kterou lze považovat za rutinní – požadavky se nemění, technologie je známá, tým dokáže „předpovídat“ (Příloha 3). Nicméně R2 je názoru, že i v tomto případě lze použít Scrum. K agilním metodikám ještě zmínil, že ne všechny firmy mají takovou kulturu nebo organizační strukturu, aby se pak mohli přizpůsobit tomu, že bude vyžadováno větší zapojení lidí, aktivní týmová práce: „nevím, jestli je to výhoda nebo nevýhoda“ (Příloha 3). Agilní týmy jsou schopné pracovat „jako jeden organizmus“, zatímco za nevýhodu waterfallu R2 označuje to, co pojmenoval „princip „tiché pošty“: kvůli malému zapojení lidí každý pracuje zvlášť, informace se nesdílují navzájem, což může být „zdrojem nedorozumění“: „V tom waterfallu se často ta informace, kterou váš zákazník skutečně potřebuje, má tendenci ztrácet [smích]. Nebo není úplně přesná, no“ (Příloha 3).

Dle R3 waterfall „ve dnešní době je neskutečně zastaralá věc...nemělo by to prostě být. Napříč prostě odvětvím to by být nemělo.“ (Příloha 4). Co se týká agilních metod, uvádí jak výhody, tak i nevýhody. Jako první výhoda byla zmíněná stoprocentní kontrola vývoje, „od počátečního nápadu až po finální produkt“ (Příloha 4). Že je možné sledovat celý proces, přičemž jak

ve Scrumu, tak i např. v Kanbanu. Druhá výhoda je zrychlení práce: „*Pokud tam jsou správné akceptační kritéria a takovy dle věci, tak je to...řeknu prostě o 60-80 procent rychlejší než cokoliv jiného*“ (Příloha 4). Za nevýhodu označuje „pomalý rozjezd“ čili přechod: učení a jiné patřičné procesy nějakou dobu trvají, což je z pohledu top-managementu problém.

Pokud lze udělat shrnutí 1. okruhu, tak je možné označit, že se respondenti shodli v pozitivním názoru na tom, co se týká přínosu a důležitosti agilních metod ve svých odpovědích na příslušné dotazy, avšak R1 je kritičtější a nemá natolik jisté hodnocení jako R2 a R3. R1 a R2 uvedli, že agilní metody mohou být přínosné pro zvýšení konkurenceschopnosti. Za výhody agilních metod byly označeny: zapojení zákazníka, poměrně snadná implementace, tým jako celek, dobrá kontrola vývoje a zrychlení práce. Jako nevýhody zmínili respondenti tyto hodnoty: problém najít lidi s odpovídajícími zkušenostmi, špatná komunikace mezi vícero týmy, potřeba náležité firemní kultury pro adaptaci agilu a pomalý přechod z waterfallu. Vodopádový model je označen jako nevhodný pro současný proces vývoje produktu (dle R3), nebo jako použitelný i dnes, ale pouze tam, kde se dá všechno předpovídat a kde jsou firemní procesy nastavené odpovídajícím způsobem (dle R2).

#### 4.2.2 Okruh 2: Scrum

##### 4. Máte zkušenosti s metodou Scrum?

R1 má obecně s agilními metodami zkušenosti jeden rok, řekl, že Scrum používají ve společnosti (navíc i Kanban), takže dá se předpokládat, že jde v podstatě o práce s touto metodikou. R2 má se Scrumem patnáctiletou zkušenost, navíc pracoval s XP. R3 se Scrumem pracoval kolem pěti let, do toho je zahrnován i Kanban.

##### 5. Uveďte prosím základní odlišení Scrumu od jiných agilních metodik.

R1 porovnal Scrum a Kanban. Uvedl, že Scrum je podle něj poměrně specifický ve svém přístupu k některým procesům (jako chování při stand-upu), Kanban je v tom jednodušší, volnější a nabízí vizualizaci, kterou se dá např. zpracovat určitý objem práce (Příloha 2). Jinak uvádí, že by jako velká firma potřebovali metodiky jako LeSS nebo SAFe, jak to bylo už zmíněno v odpovědi na otázku č. 3; inspirují se příkladem jedné holandské banky.

Dle R2 je Scrum „*moc obecný*“ (Příloha 3), i když vystihuje základní principy a je dobře definovaný.

R3 uchopil porovnání Scrumu a Kanbanu stejně jako R1 (dle jeho názoru, Kanban v podstatě ze Scrumu vychází). Jako základní odlišení Scrumu od jiných agilních metodik zmínil iterace

(sprinty), zatímco Kanban nepracuje s časem vůbec. Dodal i jiné odlišení, a to počítání velocity ve Scrumu. Jinak, dle jeho mínění, Scrum se docela podobá jiným agilním metodikám. (Příloha 4).

#### 6. Čím je možné vysvětlit takovou popularitu Scrumu?

Dle názoru R1, Scrum je tak populární díky své „nadefinované řadě“ – má přesně rozepsané, jak by měl tým vycházet procedurálně (Příloha 2). Že nejde pouze o principy, ale i o chování účinkujících, a vždy se dá získat na konci nějakého časového úseku feedback.

R2: „Myslím, že to má dva aspekty. První, že to funguje [smích].“ (Příloha 3). Druhý aspekt, že je to možné snadno aplikovat i navzdory specifické terminologii, ale pouze pokud tomu lidé rozumí. Kromě toho se zmiňuje o viditelnosti a transparentnosti – lidé vidí, co se děje, a mohou si lépe navzájem pomáhat, zároveň je obtížně něco skrýt. Kromě toho se respondent domnívá, že Scrum dnes „už je v podstatě klasika“ (Příloha 3) kvůli širokému využívání a jeho dalšímu rozvoji tak, že se propojí víc s produkcí (DevOps) a rozšiřuje se pro potřeby velkých podniků, kde je více agilních týmů.

R3 má docela kritický názor na to, proč je Scrum tak hojně využíván. „Je populární proto, že se z toho stalo novodobé klišé. <...> Dneska vlastně kdo prostě není „agile“, kdo nemá Amazon Web Services, tak je mrtvý. Samozřejmě s nadsázkou řečeno.“ (Příloha 4). Jde o to, že Scrum je tak vyžadován pouze kvůli tomu, že ho používají všichni kolem, bez pochopení, jestli je vůbec zapotřebí (pro tento podnik nebo pro určitého zákazníka).

#### 7. Napadá Vás příklad jiné agilní metodiky, která je vhodná k implementaci?

R1 doporučuje takové metodiky jako SAFe, LeSS, zmiňuje se i o Scrum of Scrums a Kanbanu. Připomíná, že existují i zvláštní techniky, jako např. TDD nebo behavior-driven design, kterých se dá využít nezávisle na tom, která agilní metodika je implementována (Příloha 2).

R2 se zaměřuje na extrémní programování a techniky, které s tím souvisí, např. kontinuální integrace a párové programování. Upozorňuje, jak i R1, na test-driven development a Kanban. Dle názoru respondenta, mohou výše zmíněné věci soužit jako doplňky Scrumu. Líbí se mu i Scrumban – dle R2 je to dobrá kombinace. Zmiňuje se o lean developmentu, že by se prvky tohoto přístupu daly také použít. R2 se domnívá, že využití doplňkových technik může značně ulehčit týmovou práci a zlepšit např. kvalitu kódu (Příloha 3). Ohledně kódu poukazuje i na přínos používání již zmíněné techniky párového programování, která pomáhá i při výměně znalostí a posílení týmu.

R3, jako i přechozí respondenti, říká o Kanbanu, ale upozorňuje, že se nehodí na všechny projekty, protože to je spíš, jak se vyjádřil R3, „*to-do-list*“ (Příloha 4), barevný a se šítky, který je možné spojit se Scrumem, ale je otázkou, jak by to šlo ve spojení s jinými metodami, což záleží na každé konkrétní společnosti.

Shrnutí ke 2. okruhu je možné udělat následující. Scrum používá každý z respondentů, i když jejich zkušenosti jsou výrazně odlišné (od jednoho roku využití až po 15 let). Názory na to, čím se liší Scrum od jiných metodik, byly poměrně různorodé: R1 se domnívá, že Scrum je docela specifický, R2 je názoru, že je příliš obecný. R1 a R3 porovnávají Scrum s Kanbanem; R1 uvádí, že Kanban je jednodušší. R3 doplňuje, že Scrum má iterace a velocity, jinak žádná zásadní odlišnosti od jiných metodik nevidí. K popularitě R1 řekl, že to je díky přesným definicím, R2 mluvil o tom, že to prostě funguje, je transparentní a poměrně snadno implementovatelný (za určitých podmínek). R3 zastupuje jiný názor – míní, že Scrum je tak populární jen díky tomu, že to je dnes prostě „trendové“.

K implementaci doporučuje R1 takové velké frameworky, jako např. SAFe; R2 se zmiňuje o XP a leanu. Všichni tři respondenti mluvili o Kanbanu. R1 a R2 se navíc zmínili o TDD a o tom, že není nutné zapomínat na jednotlivé techniky, které se dají spojit s různými agilními metodami.

#### 4.2.3 Okruh 3: Problémy při implementaci

8. Byl/a jste zaměstnán/a ve firmě během implementace agilních metod?

- a. Setkal/a jste se při implementaci agilu s nějakými problémy?
- b. Pokud ano, s jakými? Jak jste jim čelil/a?

R1 odpověděl, že byl „*součástí dodávek, kdy se ta metodika používala*“ (Příloha 2). R2 byl zaměstnán v několika firmách, kde podobná implementace probíhala, stejně je na tom i R3.

Za největší problém R1 označil začlenění agilního týmu do zbytku organizace, obzvláště pokud jde o podniková pravidla a napojení týmu na firemní procesy. Zmínil, že např. finanční řízení v jejich společnosti se dělá v podstatě dle waterfallu a tím pádem je podobné napojení do značné míry komplikované. R1 navíc mluvil o přechodu na velocity z man-dayů, že to pro ně bylo na začátku obtížné a nezvyklé. Poslední problém, který respondent uvedl, byl nedostatek místa a s tím související potíže s komunikací uvnitř týmu.

R2 byl u procesu implementace několikrát. Podle něho bylo jedním z problémů nepochopení toho, co je Scrum, což mělo vliv na proces implementace: „*Protože často někdo zaslechl*

„Scrum není žádná dokumentace“ nebo „Scrum je něco, co se dělá strašně jednoduše a nepotřebujeme žádnou architekturu“ [smích].“ (Příloha 2). Jako řešení problému vidí vzdělávání některých zaměstnanců (jen některých kvůli ceně podobné certifikace), kteří pak mohou své znalosti šířit kolem sebe. Zdůrazňuje, že je velice důležitá podpora managementu. Stanou se i nedorozumění mezi vývojářem a zákazníkem (poslední to chce podle agilních metod, pro něho to však dělají dle waterfallu). R2 doplňkově říká jako člověk, který se pracovně často pohybuje v jiném kulturním prostředí, o kulturních odliších (mentalitě), které mohou také mít značný vliv na týmovou práci.

R3, který to, co i R2, zažil více než jedenkrát, poukázal na to, že si lidé těžko zvykají na nový způsob práce nebo ne vždy chápou principy Scrumu správně (např. nezvládali systém sprintů), tím může být způsobená pomalost práce, která se nelíbí managementu (Příloha 4). Označil, že s Kanbanem podobné potíže nebyly.

9. Víte, nakolik je těžké, nebo snadné sestavit tým, který by vyhovoval agilním principům?

R1 připomíná svou odpověď na otázku č. 3, že podle něho je klíčová seniorita členů týmu. Míni, že člověk by měl mít určité schopnosti nejen pro samotný vývoj, ale i pro business nebo architekturu, a mohl by vnímat problém s přesahem (Příloha 2). Bez toho by podle názoru respondenta tým fungovat nemohl.

Dle mínění R2, to není tak těžké, záleží jen na vůli, přání spolupracovat a na chemii. Tím myslí situace, kdy jeden člověk nechce s nějakým konkrétním jedincem pracovat, ale to se dá napravit tak, že se lidé v týmech budou vyměňovat (pokud je těch týmů víc). Obecně zdůrazňuje, že je důležité, aby všichni v týmu do sebe zapadali, jak podle svých rolí, tak i podle schopností, přičemž není obzvlášť nutné, aby všichni byli skvělí programátoři, nýbrž aby lidé měli „otevřenou mysl“ (Příloha 3). Zmiňuje, že komunikace je klíčovým bodem, kvůli čemuž by neměla kapacita týmu přesahovat deset lidí. Tým pak funguje jako jeden celek a je otevřenější.

R3 se domnívá, že „není to otázka, jestli je to těžké, nebo snadné, ono je otázka, jestli jsou ty lidé ochotní v tom pracovat. Pokud jsou ty lidé **všichni**<sup>18</sup> ochotní v tom pracovat, tak je to snadný. Pokud jediný z deseti lidí není ochotný, tak je to neskutečně těžké a blíží se to k nemožnému.“ (Příloha 4). Doplňkově zmiňuje, že tým by neměl mít hodně lidí: „Když

---

<sup>18</sup> Zde a nadále platí, že slovo je zvýrazněno tučně, protože ho respondent v rozhovoru zdůraznil hlasem, což je v přepisu označeno kurzívou, avšak při citování není stejný způsob zvýraznění možné použít.

*máte 50 lidí ve scrumovém týmu, tak se můžete dostat k tomu, že jste u vodopádu – svým způsobem, jo. Prostě v chaosu.*<sup>19</sup> (Příloha 4).

10. Pokud někdo do týmu nezapadá (jsou v týmu interpersonální konflikty), jak to řešíte?

R1 odpověděl, že když práce dle agilního přístupu teprve začala, neexistovala možnost měnit lidi, takže přece jen museli nějak dohromady pracovat. Později týmy začali sestavovat na základě předchozích zkušeností, tím pádem byla snaha, aby lidé, kteří spolu nevycházejí nebo nepracují tak, jak je nutné, do projektu nevstupovali.

R2 ještě v odpovědi na otázku č. 8 řekl, že pokud jeden z členů týmu způsobuje potíže, je zapotřebí domluvit se s ním nějakým způsobem nebo ho i z týmu vyloučit. V předchozí otázce zmínil, že by bylo možné lidi mezi týmy vyměňovat. Avšak může se stát, že určitý člověk nezapadá do žádného týmu, a tam vyvstává otázka, jestli vůbec může takhle pracovat. Respondent zdůrazňuje, že „*ten tým je daleko citlivější a „diagnostický“ na takové problémy než třeba v nějakém waterfallu.*“ (Příloha 3). Navíc dodává, že obrovskou roli hraje Scrum Master, kterého porovnává s trenérem fotbalového týmu, protože se v případě problému snaží každého vyslechnout a problém vyřešit.

Dle R3 záleží na tom, jak pracuje Product Owner. Že pokud nejde o Product Ownera, jak je vnímán v agilní metodice, ale spíš o obyčejného manažera, který se snaží celý proces vývoje kontrolovat a nedává pracovníkům určitou volnost, může to celé zkazit. Ze svých zkušeností doporučuje problematické oddělení dát na outsourcing nebo ho prostě oddělit.

Shrnout výsledky 3. okruhu se dá tímto způsobem. Každý z respondentů zažil implementaci agilních metod, R2 a R3 několikrát. Každý se také setkával s obtížemi. Výčet problémů: zapojení týmu do celkem waterfallové organizace, používání velocity místo man-dayů, možný vliv prostředí (nedostatek/omezení místa) na komunikaci, nepochopení metodiky Scrum, neporozumění mezi vývojářem a zákazníkem, kulturní odlišnosti, pomalý přechod na agile a čas, který je potřebný, aby si na něj lidé zvykli. Jako řešení některých z výše uvedených problémů bylo nabízeno vzdělávání zaměstnance, což by mohlo zvýšit úroveň porozumění metodice a ulehčit přechod (implementaci). Důraz byl kladen i na potřebu podpory vyššího managementu.

K týmu a problematice s ním spojené byly názory ne ve všem stejné. R2 řekl, že sestavit tým těžké není, zatímco R3 se domnívá, že jde pouze o ochotu lidí pracovat takovým způsobem a navíc spolu. R2 také zmiňoval ochotu (přání) pracovat, doplňkově je to i o vůli a o tom, jak

---

<sup>19</sup> R3 se ani jednou během rozhovoru nezmiňoval o takových metodikách jako SAFe nebo LeSS.



k sobě lidé zapadají (jestli se nevyskytují interpersonální konflikty), – že bez splnění těchto kritérií tým fungovat tak, jak by měl, nebude. Zajímavostí je, že pokud R1 trvá na vysokých a různorodých schopnostech členů týmu, R2 se domnívá, že to tak důležité není, spíše vítá otevřený přístup k procesu vývoje.

Jako jeden ze způsobů vyřešit vnitřní konflikty uvedli všichni výměnu lidí mezi týmy. R2 navíc navrhuje promluvit si s člověkem, popřípadě ho z projektu vyloučit, a zdůrazňuje důležitost role Scrum Mastera při podobných komplikacích. R3 poukazuje na Product Ownera, který by se mohl do řešení zapojit, a navrhuje, pokud jde o nějakou týmovou jednotku, outsourcing nebo její oddělení.

#### 4.2.4 Okruh 4. Dodržování principů agilu a praxe

11. Je u Vás používán princip kolektivní viny (za chyby zodpovídají všichni členové týmu)?

R1 na tuto otázku zareagoval takto: „*Je to super princip [smích]. Používáme ho i mimo agilní týmy [smích]*“ (Příloha 2).

Odpověď R2 byla úplně opačná – přiznal se, že pojem viny nesnáší. Že tým by měl nést zodpovědnost za produkt a veškerá rozhodnutí by měla být společná. Dle jeho názoru kolektivní vina jen zvyšuje společný stres, může být zdrojem šikany, a tím škodí efektivitě (lidé dělají víc chyb). Je důležitější, aby tým prostě fungoval a soustředěně plnil svou práci, „*aby měl nějaký dobrý pocit z toho, co dělá, a když se něco nestihne, tak se prostě nestihne, jo.*“ (Příloha 3). Problém jednoho člena týmu je společný problém – podobný princip ve společnosti R2 je používán, ale dle jeho názoru by to nemělo být zneužíváno. Avšak projednání chyb může být užitečné, dle R2, pro retrospektivy a následný rozvoj týmu.

R3 ten princip zažil, ale dle jeho mínění je typické spíše hledání „obětního beránka“ (Příloha 4). Zdůrazňuje, že sám Scrum napomáhá chyby upravovat tím, že je vždy možné vrátit se zpátky a něco upravit. Domnívá se, že by ten princip neměl být používán, protože chyby dělají konkrétní lidé, a každý by měl za svůj omyl odpovídat zvlášť.

12. Jak byste popsal/a vztah Vašeho týmu se zákazníkem?

R1 uvádí, že v agilním týmu je zákazník jeho členem a klíčovým je Product Owner, který by měl umět správným způsobem stanovit priority a mít na jejich provádění pravomoc. Domnívá se, že „*...vztah s tím zákazníkem je v případě dodávek Scrumem, třeba těmi adaptivními a agilními metodami lepší, protože ten zákazník dostane ne to, co se snažil popsat na začátku,*

*ale dostane to, co chce, když je to hotové. Což je velký rozdíl “ (Příloha 2). Dle názoru respondenta problém změn agilní metodiky „určitě řeší“ (Příloha 2).*

R2 má často kvůli specifikce své společnosti zákazníka v jiné divizi. Může se stát, že kvůli tomu nemají vývojáři dostačující kontakt s businesssem, a to řeší tím, že je posílají do nějakých hubů nebo skladů, a to pomáhá lépe chápat potřeby zákazníka. Mají dokonce speciální vzdělávací program. Jiný způsob je do procesu vývoje, do týmu, dostat lidi přímo z businessu nebo z toho odvětví (jestli jde o kurýra v produktu, tak se na něho obrátit). Bylo by ideální, dle R2, mít je v týmu napořád, ale pokud to nejde, tak je minimálně zvát na sprint review, aby se k tomu vyjádřili nebo aby to i zkusili, protože Product Owner, který zastupuje zákazníka, nemá vždy natolik hluboké znalosti, jako je zapotřebí.

Úplně jinou odpověď uvedl R3. Měl zkušenost, že zákazník něco vymyslí, a to se ihned dávalo do sprintu, kvůli čemuž celý proces selhával. Dokonce respondent, který tehdy dělal Scrum Mastera, se kvůli tomu hádal s Product Ownerem. R3 se domnívá, že zákazník nemá moc velký zájem o proces, jen o výsledek jako takový. Že se podívá na releasu na to, jak to funguje (za takovou možnost respondent Scrum chválí), ale na to, jak k takovému výsledku došlo, se ptát většinou nebude.

13. Jak často se stává, že Product Owner nebo Scrum Master má pracovat s více než jediným týmem? Může to způsobit problémy v projektu?

R1 se domnívá, že Product Owner by se měl soustředit pouze na jeden projekt, protože to vyžaduje jeho neustálou pozornost, zatímco Scrum Master může vést dva-tři týmy. Ale i přesto v jejich společnosti SM chybí, kvůli čemuž je občas nabírají externě.

R2 upozoroval, že se to u nich stává poměrně často, ale dobré to podle něho není. Domnívá se, že Scrum Master by měl pracovat pouze s jediným týmem, zatímco Product Owner toho zvládne víc (dva-tři, ale maximálně pět týmů), jinak by se mohla práce výrazně zhoršit. Respondent zdůrazňuje, že se role nesmí překrývat, a že oba mají být v týmu. Scrum Master je obzvlášť důležitý v průběhu formování týmu, později nemusí být stále k dispozici, ale to neznamena, že by SM už vůbec nebyl potřebný.

R3 je rozhodně proti tomu, aby jak Product Owner, tak i Scrum Master měl na starost víc než jeden tým. Např. pokud by Scrum Master vedl několik týmů, tak by neměl ani dostatek času, ani sil věnovat se každému z nich náležitým způsobem – plánovat něco, připravovat se nebo diskutovat.

14. Vyžadujete od PO a SM nějaké certifikáty? Pokud ano, jaké?

R1 řekl, že žádné certifikáty nevyžadují, spíš je zajímá, kde se s tím lidé setkali: „*A úplně ještě ideálně, že i někdy zažili nějaké selhání toho týmu a jsou schopní se z toho poučit.*“ (Příloha 2). Mnohem důležitější jsou zkušenosti.

R2 upozorňuje, že podobné certifikace jsou moc drahé, i když velice kvalitní, ale zároveň „*ty certifikáty skutečně z toho člověka Product Ownera nebo Scrum Mastera neudělají.*“ (Příloha 3). Podle něho stačí, aby prošlo školením jen několik lidí, kteří pak budou šířit znalosti dál. Ale stejně, jako i R1, cení zkušenosti a reální životní situace. Ve společnosti mají vlastní certifikaci a tréninky.

R3, který sám několikrát dělal Scrum Mastera, odpověděl, že to nikdy od něho nevyžadovali, a ve firmách, kde pracoval, to nebylo požadováno ani od jiných lidí. Respondent vnímá certifikáty poměrně kriticky: „*K čemu to je? Upřímně, je to dvoudenní školení, na konci kterého dostanete takový kus papíru. [...] To je prostě duch moderní doby – čím víc člověk má papírů s nějakým razítkem, tím jako je lepší, což je blbost.*“ (Příloha 4). Kritizuje i ceny, za které jsou tyto certifikáty nabízeny. Podle jeho názoru to využívají lidé, kteří se ve Scrumu nevyznají a jen chtějí vydělat peníze, a pak dochází k tomu, že tato metodika ani není normálně implementována.

#### 15. Na kolika agilních projektech může zároveň pracovat jeden člověk?

R1 se domnívá, že vývojář může pracovat více než na jednom, ale spíš na dvou-třech, protože jinak člověk nebude schopen pořádně se na svou práci soustředit. Avšak zdůrazňuje, že kvůli tomu, že v jejich společnosti jsou i standartní, nejen agilní, projekty, tak takoví specialisti, jako např. business analytici, jsou vynucení pracovat i na 7–10 projektech. Samozřejmě to má vliv na kvalitu jejich práce, ale v současnosti se s tím nedá nic dělat. Podtrhuje, že na podobné věci jsou agilní týmy velice citlivé (Příloha 2).

R2 v podstatě souhlasí s tím, co řekl R1. Je důležité, aby se člověk soustředil, tím pádem je nejlepší, aby pracoval na jednom projektu, ale obvykle je tomu jinak, v jeho společnosti také. Zajímavý postřeh je, že existují role, které nejsou dle respondenta zapotřebí v týmu napořád; jako příklad R2 uvádí UI/UX designera (Příloha 3). To však neznamená, že by tým neměl být stabilní. Pokud člověk pracuje na deseti projektech, což respondent také zažil ve svém podniku ihned po implementaci agilu, tak to společnosti neprospěje, pokud ale nejde např. o takové lidi, jako testeři, které se mohou mezi projekty pohybovat.

R3 je názoru, že člověk může pracovat na několika projektech, ale závisí to na takových faktorech, jako správné naplánování sprintů, a jestli jde o práci napříč týmy nebo i firmami.

Ale zaleží i na samotném člověku, jestli to zvládne a nastaví všechno tak, aby se nezbláznil (Příloha 4). Podtrhuje, že pokud někdo něco nezvládá, tak nic se nestane.

16. Popište prosím, jak probíhá komunikace mezi členy týmu.

R1 odpověděl, že komunikace buď probíhá osobně, protože sedí v jedné kanceláři, nebo používají Slack.

Dle R2 to celkem zaleží na tom, jak je firemní kultura nastavená ve společnosti (formální nebo více neformální) a nakolik dobře plní své povinnosti Scrum Master a Product Owner. Jinak kvůli specifikaci podniku (logistika, geografické rozdělení) používají několik nástrojů, např. IBM Rational Team Concert, který je kromě jiného možné používat i na waterfallové projekty. Navíc respondent zmiňuje i Jiru.

R3 řekl, že se baví po telefonu nebo používají Slack. Zdůrazňuje, že obecně komunikace v týmu má být velice dobrá – musí se věřit a pomáhat si navzájem, a zaleží na lidech, kteří zastupují role Product Ownera a Scrum Mastera.

17. Používáte nějaký software k plánování sprintů, meetingů, pro backlogy atd.? Pokud ano, uveďte název.

V případě R1 tým používá Jiru a Trello. Jira má agilní moduly, tam tým dává backlogy a meetingy. Trello je používán pro plánování, a odtud se údaje převádí do Jiry.

R2 znovu zmínil Rational Team Concert, ve kterém je možné např. dělat backlogy. Označuje, že čím složitější nástroj je, tím vyšší flexibilitu má a tím lépe umožňuje přizpůsobit se zákazníkovi. K Jire říká, že *„Ta Jira před pěti-šesti lety byla velmi jednoduchá. Dneska se stává Jira komplexnější a komplexnější [smích]. <...> Rozhodně Jira je nejlíp integrovaný nástroj mezi dalšími produkty.“* (Příloha 3). Uvádí, že v porovnání s Rational Team Concertem je Jira dostupnější a má širší síť vendorů, ale nástroj od IBM byl od začátku vyvíjen pro koordinaci, je obecnější a má lepší integraci, např. s test managementem (Příloha 3). Avšak respondent se domnívá, že nástroj není tak důležitý – je možné úspěšně používat i zeď se stickery, ale to platí pro tým, který nemá geografické rozdělení, jinak nějaký speciální software potřebovat budou. Mohou to být i Excel nebo SharePoint.

Stanovisko R3 k Jire je docela kritické: *„Jira je hrozný paskvil, to nepoužívejte.“* (Příloha 4), ale přiznává, že je velice rozšířená. Zmiňuje Asanu, o které zaslechl, a Trello (potřebuje na Scrum speciální plugin), avšak na používání doporučuje výhradně Targetprocess. Uvádí, že je asi nejdražší ze všech výše zmíněných nástrojů (25\$/měsíčně na člověka), ale zároveň,

podle respondenta, je nejlepší; R3 s ním pracoval ve všech firmách (Příloha 4). Má spoustu služeb a širokou škálu nastavení, díky čemuž se dá přizpůsobit jakýmkoliv požadavkům. Verze zdarma má pouze tisíc entit (tasky, bugy, komentáře...), jinak se neliší od zaplacené verze. Entity je možné následně mazat. R3 označuje, že by verze zadarmo mohla stačit, pokud nejde o velkou společnost.

#### 18. Jakou délku sprintu obvykle máte?

U R1 sprint trvá dva až tři týdny. R2 měl v období, kdy se rozhovor konal, projekt, běžící na dvoutýdenních sprintech. Označil, že je to optimální varianta, jinak by délka sprintu neměla přesahovat jeden měsíc. Ale zdůrazňuje, že záleží i na samotném projektu: pokud je poměrně krátký, tak tam by se daly zkrátit i sprinty, a naopak. Úplně to samé, co i R2, říká R3 – o korelaci mezi velikostí projektu a délkou sprintu. Doplnil, že na týdenních sprintech se to nestíhalo např. testovat, a že měl takovou zkušenost, že různá oddělení (grafika a vývoj) měla i různé sprinty.

#### 19. Pokud v průběhu sprintu nějaký úkol není splněn, co se s úkolem dále stane? Jak často se Vám to stává?

Dle R1, se něco podobného stává ve více než polovině případů, že nějaký úkol není splněn. To tým buď hodí do následujícího sprintu, nebo to „řeší *bokem*“ (Příloha 2), pokud je úloha důležitá. Tým může požádat i o pomocnou roli, která s tím pomůže, ale to se neřeší Scrumem, ale spíš za použití SAFe. Proč se to vůbec stalo, se projednává retrospektivě: jestli to bylo ovlivněno okolím, dalo by se to včas překonat.

Přemístění úkolu do dalšího sprintu a diskuse na retrospektivě o tom, proč se to stalo, zmiňuje i R2. Říká, že to je typické řešení, ale zároveň zdůrazňuje, že takové přemístění úkolu ještě neznamená jeho nezbytnou implementaci, protože další sprint má už svůj backlog a tam jsou prioritní jiné stories. Dokonce může být úkol vymazán, pokud se zjistí, že není v podstatě validní. Někteří story rozdělí na hotovou část a na to, co ještě zbývá, a to poslední se přemísť jako nový úkol do dalšího sprintu, ale dle názoru R2 je lepší vnímat to, co není hotové jako celek, jako neukončené. Respondent také říká, že z podobných zkušeností, z retrospektivy by se měl tým poučit, avšak bez pocitu viny.

Reakce R3: „*Oh...tak stává se to častěji, než by mělo.*“ (Příloha 4). Stejně jako u předchozích respondentů, řešením bylo přesunutí story do dalšího sprintu. Označuje, že zkušenosti lidí a 10 až 20 sprintů pomohou nadále správně hodnotit objem práce, díky čemuž tým bude vědět, kolik toho je schopen stihnout, a nebude dávat do sprint backlogu zbytečné úkoly. Podobné situace jsou dle R3 pochopitelné a normální na začátku práce za pomoci sprintů.

## 20. Jaká je střední délka plánovacích meetingů?

R1 odpověděl, že v jejich případě jde o skoro hodinu a půl, ale je docela problematické dostat všechny lidi na tento meeting kvůli jejich rozvrhům nebo omezením s místnostmi. U R3 je zkušenost stejná, tj. hodina a půl jako střední délka.

## 21. Daily stand-up by měl ideálně být každé ráno ve stejnou dobu, ve stejném místě, ve stoje a maximálně 15 minut. Jak to máte Vy?

R1 uvádí, že se jim ve většině daří to takto dodržovat a jsou v tom i určité výhody: ve stoje lidé nemohou pasivně a uvolněně meeting sledovat – mají se do něj zapojit, a pevný čas umožňuje předem zarezervovat místnost a oznámit to lidem, aby na to byli připraveni.

R2 poukazuje, že v jejich podniku je to problematické, protože mají hodně pracovníků v Asii. Ale i když to není ve stejný čas, i tak se snaží podobný meeting uskutečňovat každý den. Dle názoru respondenta je velice důležité, aby to bylo opravdu o tom, co kdo udělal a co bude dále. Může se stát, že si na začátku adopce budou lidé myslet, že *„to je takové normální denní kafičko nebo prostě povídání, co kdo dělá a tak, jo.“* (Příloha 3). Délka záleží podle R2 na velikosti týmu, ale optimální je 15 minut. Zdůrazňuje, že pro správné vedení a chápání meetingu je velice důležitá role Scrum Mastera.

Slova R2 o tom, že vliv na délku daily stand-upu má velikost týmu, opakuje i R3. Říká, že se to dá zvládnout i za pět minut, ale *„ještě pět minut před tím všichni slézali, pět minut potom se ještě všichni nadávali [smích].“* (Příloha 4). Zažil i případ, kde se meeting jakoby přemístil do Slacku, kde lidé všechno psali, aniž by se setkávali. Respondent hodnotí meeting jako přínosný z toho pohledu, že lze takovým způsobem rychle odhalit problémy a je vidět, kdo na čem pracuje.

Okruh bude uzavřen shrnutím výše probraných otázek. Pokud začneme principem kolektivní viny, tak ve společnosti R1 se hojně používá, u R2 je také něco podobného, ale dle jeho názoru je nutné, aby to nijak neškodilo týmové práci a nezačalo to být zdrojem šikany. R3 to má za sebou, avšak na rozdíl od R1 a R2 se domnívá, že zodpovědnost by měl mít každý svou. Zákazník u R1 je v podstatě členem týmu, a tam, dle respondenta, je klíčový Product Owner, a agilní metody jsou takové spolupráci obecně velmi nápomocné. U R2 bývá zákazníkem jiná divize, což má důsledky na práci vývojářů, kteří kvůli nedostatku znalosti businessu navštěvují huby anebo např. nějaká doplňková školení. Respondent označuje, že by bylo velice přínosným zapojení někoho, kdo k tomuto produktu může mít vztah (např. pracovník skladu), který by to zkusil naživo a okomentoval jako specialista na danou oblast, což ani vývojáři, ani Product

Owner, nejsou. R3 se domnívá, že zákazníka nezajímá proces vývoje, ale jen pouhý software (produkt). Co se týká Product Ownera a Scrum Mastera, zde jsou výsledky docela odlišné. R1 zastupuje názor, že PO může vést pouze jeden tým, SM více (dva až tři). R2 má opačné mínění – že by SM měl mít na starost jeden tým, zatímco PO dva nebo tři, maximálně pět. Oproti prvním dvěma respondentům se R3 domnívá, že jak SM, tak i PO by měl mít jen jeden tým. Certifikace není v žádném ze třech případů nijak zvlášť požadovaná, mnohem cennější jsou předchozí zkušenosti, a to i negativní. R2 a R3 se shodují na tom, že žádná certifikace, která je kromě jiného i docela drahá, z pracovníka neudělá specialistu na agile.

Druhá část okruhu má zaměření na týmovou práci. R1 a R2 se shodují, že by měl člověk ideálně pracovat na jednom projektu, aby se mohl soustředit na svou práci co nejlépe, ale ve skutečnosti to tak nebývá. R2 má zkušenost s tím, že je možné do týmu připojit člověka, který by plnil nějaký úkol, a po jeho splnění odcházel (např. UI/UX designer). R3 se domnívá, že člověk může pracovat na vícero projektech, ale závisí to na takových okolnostech, jako firemní prostředí, délka sprintů projektů, plánování a rysy jednotlivého jedince. Komunikace uvnitř týmu probíhá buď osobně (R1), telefonicky (R3) nebo přes Slack (R1, R3); u R2 se používá např. IBM Rational Team Concert nebo v nějakých odděleních i Jira. R2 a R3 upozorňují, že Product Owner a Scrum Master jsou také nesmírně důležití pro zachování kvalitní, dobré komunikace mezi členy týmu. Na jiné účely jsou používány takové nástroje, jako Trello (R1, R3), Jira (R1, zmiňuje to R2, avšak R3 Jiru nedoporučuje). R2 používá IBM Rational Team Concert a říká, že jinak se dá použít cokoliv, dokonce i tabule, a na softwaru až tak nezaleží. R3 se zmiňoval o Asaně a velice doporučoval Targetprocess. Sprints mají všichni v podstatě stejné – dva týdny; mohou být odchylky do třech týdnů (R1), R2 klade důraz na to, že by délka sprintu neměla přesahovat čtyři týdny. Jak R2, tak i R3 poukazují na to, že existuje korelace mezi rozsáhlostí projektu a délkou sprintu. Nesplněné úkoly (stories) se objevují poměrně často u každého respondenta, a u všech třech je úkol ve většině případů přemístěn do dalšího sprintu. R2 uvádí, že se může stát, že úkol bude později označen za nevalidní a bude vymazán. R3 se domnívá, že podobná situace se vyskytuje častěji v týmech, které ještě nejsou tak zkušené, ale po několika sprintech budou schopny vypočítávat velocity a tím by se měla nepotřebná činnost (nebo kterou zjevně nebudou stíhat) eliminovat. Plánovací meeting má střední délku hodinu a půl u všech respondentů. Co se týká daily stand-upů, R1 to má skoro ideálně dle metodických pokynů (ale občas bývá problém s místem), u R2 to není ve stejný čas, ale snaží se meeting dělat každý den. R3 má zkušenost, že se tento meeting prováděl i přes Slack. R2 a R3 se shodují, že délka meetingu může záviset na velikosti týmu, ale neměla by přesahovat 15 minut.

## 4.2.5 Okruh 5: Výsledky implementací

22. Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily?

R1 po porovnání výsledků s waterfallem říká, že zkušenost s agilem je lepší – obzvlášť díky přímé práci se zákazníkem. Občas byl problém s nějakými specializovanými odděleními, jako např. bezpečnost nebo architektura. Ale pokud byl uživatel podle respondenta spokojený, tak samotná firma ne vždy, ale R1 doufá, že se to po implementaci robustnějších metodik (typu SAFe nebo LeSS) změní. Nic konkrétnějšího k týmu nezmiňoval.

R2 s jistotou odpověděl, že to bylo ve prospěch: zlepšila se komunikace, tým začal být více propojený. Zmínil, že občas je to na začátku problém, pokud je někdo více introvert, jak to bývá např. u programátorů.

R3 také odpověděl, že se výsledky zlepšily, ale to potřebuje čas. Po prvních sprintech to tak nebylo – jen po čtyřech iteracích se situace měnila k lepšímu. Ale respondent poukazuje na to, že zde je závislost i na délce sprintů, ale s časem tým začne odhadovat velocity přesněji.

23. Mělo zavedení agilu dopad na rozsah dokumentace k projektům?

V týmu R1 to vliv mělo bezpochybně. Specifikace pro projekty, vedeny dle waterfallu, byly mnohem rozsáhlejší, ale v agilním týmu má samotný kód vyšší prioritu než dokumentace. Ta se později dělá v Trello nebo v Jire. Avšak tam je problém, že se v tom orientuje pouze tým, ale někdo další to už tak nezvládá (co kde je možné najít atd.).

R2 se domnívá, že vliv je bezpodmínečný, avšak je zde riziko, že si někdo bude myslet, že ve Scrumu není dokumentace zapotřebí vůbec. Avšak, jak respondent poukazuje, dokumentace je nutná pro support, což je starost Product Ownera. Dokumentace může být potřebná i pro získání nějaké certifikace, nebo pro koncového uživatele (tréninkové materiály, uživatelská příručka, něco k instalaci), (Příloha 3). Navíc R2 označuje, že pokud někdo z týmu odejde a dokumentaci nezanechá, může to způsobit problém jeho zástupci, protože nebude znát kontext. Je možné dle respondenta psát dokumentaci přímo v kódu nebo vytvářet jakousi sémantickou síť pro celý produkt. Také zde hraje důležitou roli i requirement management – co, jak a kde definovat, aby s tím mohl popřípadě pracovat i zákazník. R2 říká, že to je možné dělat dvěma způsoby: z hora dolů – od obecného k detailům, nebo naopak – psát něco ihned u stories.

Jinou odpověď poskytl R3: dle něho to vliv nemělo a dokumentace by měla být součástí stories při používání jakékoliv metodiky (Příloha 4). Zaleží na tom, jestli to chce zákazník



a jestli dokumentace byla pečlivě zpracována ve společnosti i před přechodem na agile. „*Pokud ta firma dělala všechno správně před zavedením, že komentovala, tak budou komentovat i potom, pokud nekomentovali, tak nebudou komentovat ani potom.*“ (Příloha 4). R3 doporučí používat nějaké softwary, které by dokumentaci dělaly automaticky v kódu. Respondent navíc upozorňuje na přechod na GDPR, při kterém je nutné klást mnohem větší důraz na dokumentaci, než to bylo před tím.

#### 24. Bylo podle Vašeho názoru zavedení agilu přínosným pro Váš podnik?

R1: „*Prozatím myslím, že ne.*“ (Příloha 2). Vysvětluje to tím, že agilní metodiky v jejich podniku jsou poměrně málo využívány, nicméně výhody lze pocítit i v jednotlivých úsecích práce. „*Máme dvě aplikace tímhle způsobem doručené a tam se v podstatě právě podařilo tomu zákazníkovi doručit to, co on na začátku nevěděl, že chce. A v tom bylo to kouzlo.*“ (Příloha 2).

R2 se domnívá, že to rozhodně bylo přínosné, ale zmiňuje, že jejich společnost teď postupně přechází ze samotného agilu na DevOps, což by mělo pomoci spojit vývoj a support, a tím rozšířit vliv agilních metod. Dle názoru respondenta, „*...to jde i dál, že ten Scrum a agilní metodiky vývoje není základ.*“ (Příloha 3).

Odpověď R3 byla taková, že finanční přínos byl, ale ne stoprocentní, zatímco pro lidi, kteří s tím pracovali, to přínosné bylo určitě. Proces vývoje se zlepšil, stejně tak i komunikace, pracovníci se z toho učili, takže zavedení agilu mělo pozitivní následky. (Příloha 4)

Souhrn okruhu bude poměrně krátký. Dle respondentů se výsledky práce v týmu zlepšily. R1 zvláště zmiňoval spolupráci se zákazníkem, ale řekl, že je těžké sladit kontakt mezi agilním týmem a zbytkem organizace. Dle R2 a R3, kteří se v tom úplně shodují, se výsledky zlepšily určitě, obzvláště pokud jde o komunikaci a propojení uvnitř týmu, ale to dle jejich názoru potřebuje čas. Dopad na dokumentaci byl cítit v případě R1 a R2 – nebyla tak rozsáhlá, spojovala se s kódem, ale nevýhoda je podle respondentů v tom, že dokumentaci nerozuměl nikdo z vnějšku. R3 na rozdíl od R1 a R2 odpověděl, že žádný dopad nezaznamenal a uvádí, že pokud před implementací agilních metod s vedením dokumentace byl problém, zůstane s největší pravděpodobností i nadále. Přínos agilu pro podnik hodnotí všichni různě. R1 poukázal, že to zatím pro jeho společnost tak přínosné nebylo – z toho důvodu, že agile není v podniku nijak bohatě používán; domnívá se, že se to může změnit s přechodem na větší metodiky. Dle R2 to přínosné bylo; zároveň se v jeho organizaci řeší pokročilejší používání agilu, a to ve spojení s DevOpsem. R3 označuje, že pro firmu to moc přínosné nebylo, ale pro lidi určitě ano (osobní vývoj, vzdělávání, lepší komunikace).

#### 4.2.6 Okruh 6: Hodnocení a doporučení

25. Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.

Dle R1 je největší dopad v tom, že zákazník během procesu vývoje produktu může svou představu onoho produktu upřesnit. Stává se, že na začátku žádná konkrétní vize není, ale postupně se projekt testuje a po feedbacku to může zákazník (koncový uživatel) zkorigovat, což, podle slov respondenta, „...*je ta ohromná výhoda toho iterativního procesu*“ (Příloha 2).

R2 souhlasí s tím, že výhoda je ve spolupráci se zákazníkem, který dostává to, co doopravdy chce – zajišťuje a optimalizuje se přidaná hodnota produktu a právě to je dle respondenta to nejdůležitější (Příloha 3).

R3 uvádí seznam výhod: zlepšení výsledků a stylu práce, zvýšení rychlosti práce, hezčí kód. Označuje, že výhodou je i to, že výsledky jsou „hmatatelné“ – nejde o něco empirického, ale o skutečné výsledky, které jsou vidět za poměrně krátký časový úsek. (Příloha 4)

26. Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?

Ve společnosti R1, dle jeho názoru, nejsou agilní týmy využity tak, jak by mohli, a to i kvůli nedostatku lidí; navíc zde není sladění, koordinace mezi týmy a celou firmou. Druhá věc je to, že např. holandská banka, o které se respondent už zmiňoval dříve, po dvou letech zaznamenala nějakou anarchii, protože lidé v některém smyslu nepochopili práci dle tohoto přístupu správným způsobem; R1 tím míní problém jak řízení týmů, tak i správného porozumění agilním principům. Třetí negativní dopad byl takový, že někteří lidé nedokázali najít své místo, svou roli v novém prostředí a ztratili motivaci; někdo dokonce i odešel. „*Takže si myslím, že nestačí jenom ta správná metodika, dostatečně robustní, ale je to opravdu transformační změna. Lidé musí pochopit tu změnu. Je to těžký a dlouhotrvající proces.*“ (Příloha 2).

R2 se domnívá, že agilní metody potřebují určitou disciplínu a že je otázkou, je-li společnost, která se do toho chystá, schopná to provést. Pokud jde o obrovský podnik, dobře fungující dle waterfallu, tak R2 připouští, že by společnost mohla být efektivní, používajíc i takových metod řízení. Avšak co se týká novějších technologií, respondent nevidí žádnou překážku v používání Scrumu.

R3 uvádí několik nevýhod, a to pomalý start (kvůli čemu lidé nebyli spokojeni), lhostejnost některých zúčastněných (což však, dle mínění respondenta, může být důsledkem předchozího bodu – někdo už nechtěl takto pracovat).

## 27. Co byste doporučoval/a lidem, kteří chtějí dané metodiky implementovat?

Dle R1 je nejdůležitější, aby lidé pochopili, proč tu metodiku vůbec chtějí implementovat, jestli jim to doopravdy pomůže firemní procesy změnit k lepšímu, a jestli si uvědomují, že to nebude snadné – agilní metody mohou přinést i své problémy, které jsou řešitelné, ale je otázkou, zda to bude pro společnost přínosné. Ještě v odpovědi na 1. otázku respondent řekl, že pokud by firmě její současný stav nevyhovoval, možná by bylo načase zamyslet se nad tím, jak to změnit, aby to poskytlo co nejvíc výhod. Pokud jsou v podniku všechny procesy nastavené dobře a žádný problém se řízením produktů nevyskytuje, varuje vzhledem k velké popularitě Scrumu, že „...jít do toho jen proto, že je to moderní, – to by mi nedávalo moc smysl.“ (Příloha 2). Označuje, že podle jeho usuzování dnes mohou být agilní metodiky vhodné pro implementaci, aspoň pokud jde o bankovní sektor. Domnívá se, že v současnosti jsou banky více méně podobné v tom, co se týká produktů. „*Tam, kde se snažíme vzájemně odlišit, je, jak ty produkty nabízíme.*“ (Příloha 2). Pro další digitální rozvoj jsou také vhodné dle respondenta agilní metodiky, nebo lean development. Zároveň však R1 dospívá k názoru, že existují odvětví, jako např. průmyslové, kde se nasazení agilních metodik nehodí. Uvnitř společnosti, kde R1 pracuje, není z jeho pohledu agile vhodný pro bankovní backendy (realizace plateb, finanční operace), protože jde o poměrně komplexní, robustní systém (Příloha 2).

R2 se domnívá, že hlavní je mít „otevřenou mysl“ a doporučuje se vyškolit, v ideálním případě zúčastnit existujícího projektu (Příloha 3). Připouští, že bude těžké změnit hierarchickou strukturu, ale je přesvědčen, že to jde, obzvlášť pokud mají podporu managementu, který hraje důležitou roli, a pokud to není z donucení. Myslí, že je užitečné pochopit to, jak Scrum funguje, přes praktický zážitek, a nepodceňovat týmové role a pravidla této metodiky, protože by jinak mohl nastat chaos. Mít definition of done, provádět meetingy, připravovat se, nastavit procesy správným způsobem, dodržovat všechny pokyny. „*A myslím si, že jako to nevzdávat. Po nějakém proudným nadšení přichází takový propad, když zjistí, že to bolí, jo. A v ten moment prostě to nevzdat [smích].*“ (Příloha 3). Utrpení a práce se vyplatí tím, že se produktivita a efektivita vývoje zlepší.

R3 říká, že pokud by takové doporučení měl dát např. CEO nebo řediteli vývoje, tak by je poprosil zůstat v klidu a nedělat to jen kvůli nějakému tlaku. Navíc je nutné metodám rozumět (jak to funguje, kde začít a jak do toho jít správně), jinak budou výsledky poměrně špatné.

Důležité je, aby to chápali i zákazníci, co pro ně agile bude znamenat z pohledu dodávání produktu. Co se týká týmu, „...*tak to, jak bylo v nějakém filmu – musí vydržet*” (Příloha 4). Dle ujištění R3 začátek snadný nebude a tým by se měl držet svého a v tom pokračovat – dělat sprinty a kontrolovat je za pomoci takových technik, jako např. burndown graf. „*Jestli fakt mám doporučit někomu něco, tak prostě vydržet. Opravdu stojí to za to.*“ (Příloha 4).

28. Chybělo Vám něco v našem dotazníku? Chtěl/a byste jej nějak doplnit?

R1 v průběhu rozhovoru řekl, že ho prozatím nic nenapadá, ale později napsal e-mail (Příloha 5), ve kterém uvedl, že v dotazníku nebyla prozkoumána taková problematika, jako práce firem po transformaci (přechod z waterfallu na agile) – jestli se společností podařilo nastavit agile tak, jak má být, a jestli se realizace skutečně povedla po všech stranách, tj. že např. v týmu funguje naplno Product Owner, který vlastní product backlog a umí provádět prioritizaci úkolů. Mluví o tom, že se často setkává s mícháním waterfallu a agilu, např. po waterfallové analýze jde agilní implementace, a že to může být způsobeno tím, že se transformace často dělá jen v jednom oddělení, obvykle v IT, ale je to nutné provést v celé firmě.

Dle názoru R2 je nutné zaměřit se i na zkoumání toho, jak se dá koordinovat práce agilních týmů ve větších podnicích. Scrum to sám neřeší. Zmiňuje jak SAFe, tak i např. Discipline Agile Delivery, který je podle respondenta ještě slibnější. Možná by se dle R2 daly prozkoumat i vazby mezi agilním týmem a projektovým managementem. Uvádí, že to je občas problém, protože ve Scrumu role projektového manažera není, ale podle analýzy, kterou ve společnosti R2 udělali, v rozporu např. s Prince 2 implementace Scrumu není. Kromě toho Scrum neřeší i management rizik. Doplnkově by se dalo zaměřit na agile a automatizace, která je v současnosti všudypřítomná – jak se to řeší a jaký to má vliv na práci týmu, jak to zvyšuje kvalitu produktu a vztah se zákazníkem.

R3 chybělo zkoumání velocity a to, jak se úspěšnost agilního týmu vypočítává – jakými způsoby, jestli je to např. burndown graf nebo něco jiného.

Shrnutí okruhu je následující<sup>20</sup>. Respondenty za pozitivní dopady označili spolupráci se zákazníkem, feedback z jeho strany a včasná korekce požadavků (R1, R2); zlepšení výsledků, kódu, stylu a rychlosti práce (R3). Jako negativní dopady byly poznamenány pomalý start (R3), problematická koordinace agilního týmu s jinými odděleními, pokud poslední pracují dle waterfallu (R1); odchod lidí, kteří postrádají motivaci a přání takovým způsobem pracovat (R1, R3); potřeba určité disciplíny, kvůli nedostatku může nastat anarchie (R1, R2); nutnost mít

---

<sup>20</sup> Odpovědi na otázku č. 27, tj. doporučení, zde zahrnuté nebudou, protože doporučení budou součástí závěru.

schopnosti k tomu, aby bylo možné tyto změny ve firmě provést (R2). Všichni respondenti se shodují, že častým problémem je neporozumění agilním metodám a tomu, jak fungují.

Každý respondent poukázal na to, že dle jeho názoru, v dotazníku chybí některé důležité věci k prozkoumání. R1 by dodal otázky, které se týkají stavu společnosti po její transformaci: jestli se všechno povedlo, jsou-li role a principy dodržovány. Také by se zaměřil na míchání waterfallu a agilu – dle jeho zkušeností se to stává tam, kde různá oddělení používají různé metodiky (agile není implementován v celém podniku). Dle R2, by bylo vhodné prozkoumat koordinaci týmů uvnitř velké společnosti, a to stejně jako i řízení rizik nebo vyjednávání s projektovým managementem, protože Scrum to všechno neřeší. Navíc by respondenta zajímalo propojení agilu a automatizace – je-li to pomocné pro členy týmy, a pokud ano, jakým způsobem. R3 by se více zaměřil na měření úspěšnosti týmu, s pomocí kterých metodik to organizace dělají (velocity, burndown graf atd.).

Výše uvedené návrhy na změnu (zlepšení) dotazníku poukazují na to, že výzkum lze rozšířit na další okruhy, které by se měly zabývat post-transformačním osudem společností, metodami pro velké podniky (SAFe, LeSS) a použitím nových technologií ve spojení s agilními metody, a to např. Robotic Process Automation (RPA). Podobné úvahy mohou posloužit jako inspirace pro další výzkumy, a to nejen se zacílením na Českou republiku.

#### 4.2.7 Frekvenční analýza textů

Pro tento typ analýzy byly použité speciální stránky (<http://text.stinolez.com/>)<sup>21</sup>, které nabízejí možnost zanalyzovat text jakékoliv délky, a to jak v češtině, tak i v angličtině. Jiný vhodnější nástroj nebyl pro daný účel nalezen. Texty každého rozhovoru byly postupně zkopírovány a analyzovány systémem, který nám nabídl následující výsledky.

První rozhovor (R1, Příloha 2):

- Celkem se vyskytuje 4374 slov, z toho unikátních – 1453;
- Pokud nebudeme brát v potaz frekvenci zájmen, spojek a jiných bezvýznamových slov (jako „to“, „že“, „tak“ atd., které patří mezi 25 nejpoužívanějších slov), tak uvidíme, že nejpoužívanějším podstatným jménem je „tým“ (33. místo, počet výskytů – 20 (0.46 %), tvar „týmu“). Dále následuje „problém“ (34. místo, počet výskytů – 19 (0.43 %)).

---

<sup>21</sup> Poznámka autora: pokud by čtenář chtěl nástroj zkusit samostatně, ať s ním pracuje přes vyhledavač Google Chrome. Mozilla Firefox neumožňuje pracovat se statistickými výsledky.

Druhý rozhovor (R2, Příloha 3):

- Vyskytuje se 10433 slov, z toho unikátních – 2512;
- Nejpoužívanějším podstatným jménem je „tým” (22. místo, počet výskytů – 69 (0.66 %)). Dále je „scrum” (25. místo, počet výskytů – 64 (0.61 %)).

Třetí rozhovor (R3, Příloha 4):

- Text obsahuje 6517 slov, z toho unikátních – 1783;
- Nejpoužívanějším podstatným jménem je „scrum” (33. místo, počet výskytů – 25 (0.38 %)). Dále následují „lidé” (43. místo, počet výskytů – 22 (0.34 %)).

Agilní metodiky (např. Scrum) jsou poměrně hodně závislé na jednotlivých lidech, týmy jako organizační složka jsou často považovány za „kámen úrazu“ neboli problém.

Použitý nástroj není ideální pro analýzu textu, avšak stejně může nabídnout některé zajímavé postřehy k zamyšlení.

## Závěr a doporučení

Agilní metody se šíří světem, i Česká republika není výjimkou. Různorodé organizace se snaží implementovat agile se záměrem zlepšit zisk, zvýšit svou konkurenceschopnost a stát se flexibilnější, avšak na příkladu jak výsledků případových studií, tak i rozhovorů s lidmi, kteří s agilními metodami nějakou dobu pracují, se ukazuje, že cesta od záměru implementovat (adoptovat) dané metodiky a techniky, až po jejich skutečné – a úspěšné – využití, není snadná.

Jak na základě provedené analýzy odpovídající literatury, tak i za pomoci rozhovorů s třemi respondenty, byly definovány některé důvody firem k implementaci agilních metod, pozitivní a negativní dopady a možné budoucí cesty společnosti – jak v práci dle agilu pokračovat. Navíc lze navrhnout určité doporučení k implementaci agilních metod, které by mohlo být přínosné pro patřičné zájemce, což je hlavním cílem diplomové práce.

Pokud vedení organizace všechno pečlivě zvážilo a opravdu chce přejít na agilní metody, je možné tomuto podniku dát následující doporučení:

- Je nutné chápat, proč do toho jdete. Pokud kvůli tomu, že je to teď trendové, tak daný záměr nedává smysl už od začátku;
- Pokud taková možnost existuje, vyplatí se zúčastnit nějakého projektu;
- Promluvte se zákazníky, týmy a všemi dotyčnými, aby každý věděl, co jsou agilní metody zač a kvůli čemu se přechod koná;
- Popište zákazníkovi, co se nyní změnilo (důležité je popsat role Product Ownera) a jaké výhody z toho může těžit;
- Implementaci je možné provádět postupně a jak poukázal Tripp (2016, s. 176–177), je lepší začít od nejtěžšího místa;
- Před zahájením práce udělejte školení týmů, je možné si najmout i kouče, který by na začátku proces sledoval a pomáhal zvyknout zaměstnancům na nový styl práce;
- Tým by měl pečlivě vybrat jak samotnou metodiku (nebo spojení takových), tak i techniky. Zaměřte se na to, co chcete zlepšit, a tím se řiďte;
- Vyberte a nastavte ukazatele pro měření úspěšnosti týmové práce (velocity, burndown graf, key performance indicator atd.);
- Každý člen týmu by měl mít motivaci pracovat dle agilních metod. Mluvte s týmem a zaříd'te šikovného Scrum Mastera a Product Ownera;
- Respondenti, stejně tak i několik autorů (MOE, 2010; PIKKARAINEN, 2017; SENAPATHI, 2012), poukázali na to, že implementace nebude úspěšná bez podpory

vyššího managementu. Podporu budou potřebovat jak samotné týmy, tak i Scrum Master a Product Owner;

- Vzdělávání je proces, který bude neustále zapotřebí. Pokud chcete, aby implementace byla úspěšná, dbejte na tom, aby se zvyšovala úroveň každého člena týmu, obzvlášť technická;
- Dodržujte pravidla, role a určitou disciplínu, jinak může nastat situace, že proces vývoje nebude kontrolován v potřebné míře;
- Pokud jsou v týmu problémy s komunikací, dá se využít spojení Scrumu s takovou technikou XP, jako párové programování, které by mohlo posílit nejen komunikační schopnosti, ale i technické znalosti jednotlivců (MOE, 2010, s. 489);
- Pokud má společnost několik týmů a používá např. SAFe, lze vytvořit pro lepší koordinaci speciální tým, k tomu určený, který by byl v hierarchii výše a pomáhal by ostatním (HEIKKILÄ, 2017, s. 2925);
- Nastavte řízení požadavků, popřípadě použijte waterfall (HEIKKILÄ, 2017, s. 2919–2920).

Jak řekl každý respondent z provedených rozhovorů pro diplomovou práci, snadné to nebude a pro proces implementace, ať částečné nebo ne, je potřebný čas, rozpočet a trpělivost. Agilní metody mají své výhody a nevýhody, avšak při rozumném postupu a s doopravdovým přáním změnit jak podnikové procesy, tak i finální produkt, se to vyplatí.

Některá doporučení byla definována, avšak v budoucnu by se dal původní výzkum rozšířit na některé další otázky, jako např. spojení agilu a automatizace, a bylo by možné prozkoumat oblasti spojené s post-implementačním vývojem společností. Nicméně tato práce může posloužit jako inspirace pro další výzkumy, protože problematika agilních metodik bude i nadále perspektivní a aktuální.



## Seznam použitých zdrojů

ABBAS, Noura, Andrew M. GRAVELL a Gary B. WILLS, 2008. Historical Roots of Agile Methods: Where Did “Agile Thinking” Come From?. In: *Agile Processes in Software Engineering and Extreme Programming: 9th International Conference, XP 2008, Limerick, Ireland* [online]. Berlin: Springer, s. 94-103 [cit. 2018-03-26]. ISBN 978-3-540-68255-4. Dostupné z: <https://link.springer.com/content/pdf/10.1007%2F978-3-540-68255-4.pdf>.

Agile Practices Timeline. *Agile Alliance* [online]. [2018] [cit. 2018-03-25]. Dostupné z: <https://www.agilealliance.org/agile101/practices-timeline/>.

ANTLOVA, Klara, 2014. Agile Approach in the Project Management of the Czech Companies. In: *Procedia Technology* [online]. **16**, 929-933 [cit. 2018-04-07]. ISSN 2212-0173. Dostupné prostřednictvím Science Direct z: <https://doi.org/10.1016/j.protcy.2014.10.045>.

BECK, Kent et al., 2001. *Manifesto for Agile Software Development* [online]. [cit. 2017-10-14]. Dostupné z: <http://agilemanifesto.org/iso/cs/manifesto.html>.

HEIKKILÄ, Ville T., Maria PAASIVAARA, Casper LASSSENIUS, Daniela DAMIAN a Christian ENGBLOM, 2017. Managing the requirements flow from strategy to release in large-scale agile development: a case study at Ericsson. In: *Empirical Software Engineering* [online]. **22**(6), 2892-2936 [cit. 2018-04-03]. ISSN 1573-7616. DOI: <https://doi.org/10.1007/s10664-016-9491-z>.

HENRIKSEN, André a Sven Arne R. PEDERSEN, 2017. A qualitative case study on agile practices and project success in agile software projects. In: *Journal of Modern Project Management* [online]. **5**(1), 62-73 [cit. 2018-03-25]. ISSN 2317-3963. Dostupné z: <http://www.journalmodernpm.com/DOI/JMPM01306.pdf>.

MENS, Tom, 2008. Introduction and Roadmap: History and Challenges of Software Evolution. In: *Software Evolution* [online]. Berlin: Springer, s. 1-11 [cit. 2018-03-28]. ISBN (online) 978-3-540-76440-3. Dostupné z: <https://link.springer.com/content/pdf/10.1007%2F978-3-540-76440-3.pdf>.

NYCE, Caroline Mimbs, 2017. The Winter Getaway That Turned the Software World Upside Down: How a group of programming rebels started a global movement. In: *The Atlantic* [online]. [cit. 2018-03-26]. Dostupné z: <https://www.theatlantic.com/technology/archive/2017/12/agile-manifesto-a-history/547715/>.

MOE, Nils Brede, Torgeir DINGSØYR a Tore DYBÅ, 2010. A teamwork model for understanding an agile team: A case study of a Scrum project. In: *Information and Software Technology* [online]. **52**(5), 480-491 [cit. 2018-03-28]. ISSN 0950-5849. Dostupné prostřednictvím Science Direct z: <https://doi.org/10.1016/j.infsof.2009.11.004>.

MYSLÍN, Josef, 2016. *Scrum: průvodce agilním vývojem softwaru*. 1. vydání. Brno: Computer Press. ISBN 978-80-251-4650-7.

O Nás, [2018]. *Agilní asociace: Sdružení příznivců Agilních metod* [online]. [cit. 2018-02-23]. Dostupné z: <https://agilniasociace.cz/about/>.

PIKKARAINEN, Minna, Outi SALO, Raija KUUSELA a Pekka ABRAHAMSSON, 2012. Strengths and barriers behind the successful agile deployment – insights from the three software intensive companies in Finland. In: *Empirical Software Engineering* [online]. **17**(6),

675-702 [cit 2018-03-29]. ISSN 1573-7616. DOI: <https://doi.org/10.1007/s10664-011-9185-5>.

RIGBY, Darell K., Jeff SUTHERLAND a Hirotaka TAKEUCHI, 2016a. Embracing Agile. In: *Harvard Business Review* [online]. [cit. 2018-03-26]. ISSN 0017-8012. Dostupné z: <https://hbr.org/2016/05/embracing-agile>.

RIGBY, Darell K., Jeff SUTHERLAND a Hirotaka TAKEUCHI, 2016b. The Secret History of Agile Innovation. In: *Harvard Business Review* [online]. [cit. 2018-03-26]. ISSN 0017-8012. Dostupné z: <https://hbr.org/2016/04/the-secret-history-of-agile-innovation>.

SENAPATHI, Mali a Ananth SRINIVASAN, 2012. Understanding post-adoptive agile usage: An exploratory cross-case analysis. In: *Journal of Systems and Software* [online]. **85**(6), 1255-1268 [cit. 2018-03-30]. ISSN 0164-1212. Dostupné prostřednictvím Science Direct z: <https://doi.org/10.1016/j.jss.2012.02.025>.

SERRADOR, Pedro a Jeffrey K. PINTO, 2015. Does Agile work? – A quantitative analysis of agile project success. In: *International Journal of Project Management* [online]. **33**(5), 1040-1051 [cit. 2018-02-14]. ISSN 0263-7863. Dostupné prostřednictvím Science Direct z: <https://doi.org/10.1016/j.ijproman.2015.01.006>.

SCHWABER, Ken a Jeff SUTHERLAND, 2017. *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game* [online]. [cit. 2018-02-24]. Dostupné z: <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>.

STOICA, Marian, Bogdan GHILIC-MICU, Marinela MIRCEA a Cristian USCATU, 2016. Analyzing Agile Development – from Waterfall Style to Scrumban. In: *Informatică Economică* [online]. **20**(4), 5-14 [cit. 2018-02-11]. ISSN 1842-8088. DOI: <https://doi.org/10.12948/issn14531305/20.4.2016.01>.

ŠOCHOVÁ, Zuzana a Eduard KUNCE, 2014. *Agilní metody řízení projektu*. Brno, Computer Press. ISBN 978-80-251-4194-6.

TOMÁNEK, Martin, 2015. Současný stav používání agilních metodik ve světě a v ČR. In: *Acta Informatica Pragensia* [online]. **4**(1), 4-17 [cit. 2018-02-23]. ISSN 1805-4951. DOI: <https://doi.org/10.18267/j.aip.48>.

TRIPP, John F. a Deborah J. ARMSTRONG, 2016. Agile Methodologies: Organizational Adoption Motives, Tailoring, and Performance. In: *Journal of Computer Information Systems* [online]. **58**(2), 170-179 [cit. 2018-04-04]. Online ISSN 2380-2057. DOI: <https://doi.org/10.1080/08874417.2016.1220240>.

VARHOL, Peter, 2015. To agility and beyond: The history – and legacy – of agile development. In: *TechBeacon* [online]. [cit. 2018-23-24]. Dostupné z: <https://techbeacon.com/agility-beyond-history%E2%80%94legacy%E2%80%94agile-development>.

VersionOne, 2017. *11th Annual State of Agile Report* [online]. VersionOne Inc [cit. 2018-03-23]. Dostupné z: <https://explore.versionone.com/state-of-agile/versionone-11th-annual-state-of-agile-report-2>.

VITÁSEK, Tomáš, 2018. *Text Alchemy* [online]. [cit. 2018-04-23]. Dostupné z: <http://text.stinolez.com/>.

## Seznam obrázků

Obrázek 1. Agilní metodiky a praktiky v roce 2017 dle VersionOne. Zdroj: VersionOne, 2017, s. 10 35

Obrázek 2. Agilní techniky v roce 2017 dle VersionOne. Zdroj: VersionOne, 2017, s. 10.....36

# Seznam příloh

Příloha 1. Reflexe agilních metod – otázky k rozhovoru

Příloha 2. Přepis rozhovoru s respondentem č. 1

Příloha 3. Přepis rozhovoru s respondentem č. 2

Příloha 4. Přepis rozhovoru s respondentem č. 3

Příloha 5. Kopie e-mailu R1 z 8.12.2017

# Přílohy

## Příloha 1. Reflexe agilních metod – otázky k rozhovoru

### Obecné (úvod)

1. Připadají Vám agilní metody přínosné, nebo ne?
2. Proč je dnes důležité se zabývat agilními metodami?
3. Jaké výhody a nevýhody mají podle Vašeho názoru agilní metody v porovnání s waterfallem?

### Scrum

4. Máte zkušenosti s metodou Scrum?
5. Uveďte prosím základní odlišení Scrumu od jiných agilních metodik.
6. Čím je možné vysvětlit takovou popularitu Scrumu?
7. Napadá Vás příklad jiné agilní metodiky, která je vhodná k implementaci?

### Problémy při implementaci

8. Byl/a jste zaměstnán/a ve firmě během implementace agilních metod?
  - a. Setkal/a jste se při implementaci agilu s nějakými problémy?
  - b. Pokud ano, s jakými? Jak jste jim čelil/a?
9. Víte, nakolik je těžké, nebo snadné sestavit tým, který by vyhovoval agilním principům?
10. Pokud někdo do týmu nezapadá (jsou v týmu interpersonální konflikty), jak to řešíte?

### Dodržování principů agilu a praxe

11. Je u Vás používán princip kolektivní viny (za chyby zodpovídají všichni členové týmu)?
12. Jak byste popsal/a vztah Vašeho týmu se zákazníkem?
13. Jak často se stává, že Product Owner nebo Scrum Master má pracovat s více než jediným týmem? Může to způsobit problémy v projektu?
14. Vyžadujete od PO a SM nějaké certifikáty? Pokud ano, jaké?
15. Na kolika agilních projektech může zároveň pracovat jeden člověk?
16. Popište prosím, jak probíhá komunikace mezi členy týmu.
17. Používáte nějaký software k plánování sprintů, meetingů, pro backlogy atd.? Pokud ano, uveďte název.
18. Jakou délku sprintu obvykle máte?

19. Pokud v průběhu sprintu nějaký úkol není splněn, co se s úkolem dále stane? Jak často se Vám to stává?
20. Jaká je střední délka plánovacích meetingů?
21. Daily stand-up by měl ideálně být každé ráno ve stejnou dobu, ve stejném místě, ve stoje a maximálně 15 minut. Jak to máte Vy?

### **Výsledky implementaci**

22. Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily?
23. Mělo zavedení agilu dopad na rozsah dokumentace k projektům?
24. Bylo podle Vašeho názoru zavedení agilu přínosným pro Váš podnik?

### **Hodnocení a doporučení**

25. Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.
26. Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?
27. Co byste doporučoval/a lidem, kteří chtějí dané metodiky implementovat?
28. Chybělo Vám něco v našem dotazníku? Chtěl/a byste jej nějak doplnit?

## **Příloha 2. Přepis rozhovoru s respondentem č. 1**

Jsem ve firmě XY jako enterprise architekt. S agilními metodami pracuji osobně každý rok. A je to vždycky tak, že jsem byl třeba na dvou, třech projektech, které takhle fungovaly. A z toho roku to je v podstatě 5–10 % kapacit. Takže pro nás vlastně, když jsem zmiňoval, že jsem enterprise architekt, tak my jako architektura s tím agilním přístupem máme ten problém, jak jsem vám říkal. Že jsme vlastně prozatím nenašli mechanismus, jak ty agilní týmy, které fungují nějak autonomně, jak je napárovat s těmi principy a patterny, které se snažíme prosazovat. Proto jsme to řešili třeba i tím nasazením člověka přímo do toho, aby on tam přinesl tu znalost a potřebu. A právě vzhlížíme k té metodice Scaled Agile for Enterprise, která toto řeší.

### **Obecné (úvod)**

#### **1. Připadají Vám agilní metody přínosné, nebo ne?**

Ony mají svoje výhody, mají i svoje nevýhody. Záleží na tom správném použití, takže podle mého ta správnost toho nasazení té metodiky dané je to, co potom z toho dělá tu přínosnou metodu. Nám se daří tady většinou v XY, to zacílit správným způsobem, takže přijdou nám přínosné, i když my jsme už párkrát taky spálili s tím, že jsme nedokázali vlastně z toho vytěžit ty výhody a spíš jsme těžili ty nevýhody, a pak to rozhodně přínosné nebylo. [...] Já bych teda tu změnu dělal jenom tehdy, pokud' já jsem v nějakém stavu, kdy ten stav mi nevyhovuje. Mám nějaký problém a potřebuju ho řešit. Takže já si vlastně zvolím jinou metodu, kde převáží ty výhody nad těma nevýhodama. Takže pokud' jsem ta firma, která nemá problém s tím změnovým řízením těch svých produktů, které poskytuje, tak jít do toho jen proto, že je to moderní, – to by mi nedávalo moc smysl. Proto si myslím, že tam bude asi u řady firem odpor k té změně. Banky obecně mají problém v tuhle chvíli. Proto vymýšlejí hore dolem [způsoby], jak změnit jinak ty produkty, které poskytují, ale i jak změnit ty procesy, které fungují, protože problém pro bankovníctví ve světě... Kolem nás číhá nebezpečí v podobě malých firem, těch takových startupů, které jsou právě rychlé, a my si toho uvědomujeme, a proto se potřebujeme měnit ty procesy, a proto se zabýváme těmi agilními metodikami.

#### **2 Proč je dnes důležité se zabývat agilními metodami?**

[Odpověď je zahrnutá v předchozím bodu]

#### **3 Jaké výhody a nevýhody mají podle Vašeho názoru agilní metodiky v porovnání s waterfallem?**

Já osobně, teda to bude můj subjektivní pohled na věc...Chtěl bych upřesnit na začátek, že nemám nějakou robustní znalost tý teorie, tohoto teoretického frameworku. Ale z toho, jak to používáme, tak fungování těch agilních metodik v našem případě se opírá o malé týmy, které v podstatě řeší realizaci té změny adaptivním způsobem. K tomu, aby ta adaptace v průběhu té implementace byla možná, tak ten tým musí být sestavený ze seniorních lidí. Nemůžou tam být jakoby junioři, nebo jako průměrní hráči. Bude to velké riziko v použití těch agilních metodik. Prostě, když si vezme člověk k distribuci schopnosti, jak jsem říkal, v IT XY máme tisíc lidí a k distribuci těch schopnosti jen nějakých 20–30 % lidí je schopné vlastně plnit tu roli té seniority, mít tu úroveň té seniority, takovou, aby mohli fungovat v agilním týmu. A teď, když banka přemýšlí, že ta pomalost té změny je našim problémem a že udělá vysloveně strukturální změnu na ty agilní metody kompletně, tak já tam vidím různá velká rizika. Že nebudeme schopní najít lidi, kteří by fungovali, a vlastně juniorita těch lidí zruinuje výhodu těch agilních metod.

## Scrum

### 4 Máte zkušenosti s metodou Scrum?

Metodu Scrum my tady používáme a je to... Vlastně teď nevím, jak povídat o metodě Scrum, protože já tam jako vidím...ve firmě, jako je naše,, kdy lidé...jsou stovky lidí, kteří vlastně fungují na těch dodávkách, tak použití samotné metody Scrum v našem prostředí činí problémy. To je metoda, která se fokusuje jen na daný tým, a neřeší tu spolupráci týmů mezi sebou. Takže já tam vidím hodně těch nevýhod. Že my tady máme v současné době nějakých pět, šest týmů, které fungují ve Scrumu. S čím se potýkáme, je, že ta metodika jako taková neposkytuje nějaký návod k tomu, jak řídit tu koordinaci mezi týmy. Že ta samotná... Teď se nebavíme o takových dalších metodikách jako Scaled Agile for Enterprise nebo LeSS a podobné, které už řeší tu koordinaci. Tak potom ty týmy fungují dost autonomně a někdy se ta autonomie až zaměňuje za anarchii a tam máme hrozný problém.

### 5 Uved'te prosím základní odlišení Scrumu od jiných agilních metodik.

My jako velká firma potřebujeme metodiky, které řeší, jak týmy mezi sebou *založnovat*, jakým způsobem, buď to SAFe, nebo LeSS, nebo další metody, které asi existují. Třeba pro nás v bankovníctví velkým příkladem je ING banka v Holandsku, která udělala tu změnu, že kompletně v podstatě jak IT, tak centrálu, tak pobočky změnila organizační struktury na agilní metody. A ony se vyvinuly i nějakou další svou metodiku, která právě jakoby řeší tu koordinaci těch malých týmů mezi sebou. Nevím, co XY použije. Jedna věc je, že kromě



Scrumu koukáme hodně po těch velkých frameworkích, a druhá věc je, že v některých případech používáme Kanban, metodiku...jenom, když potřebujeme jednodušší akorát přístup k těm věcem, kde nám jakoby nezáleží na atributech chování při stand-upu, a toho, kde je Scrum hodně specifický. Stačí nám akorát vizualizace toho objemu práce, který se pro daný sprint má řešit...a touhle metodu zpracovávat...tak tam používáme občas Kanban. A používáme k tomu třeba i nástroj Trello. On vlastně hodně podporuje šítkama tu kanbanovou mapu.

#### 6 Čím je možné vysvětlit takovou popularitu Scrumu?

Přijde mi, že je nejpopulárnější. Možná je to tím, že opravdu má nadefinovanou řadu. Ta metodika je hodně podrobná v tom, jak ten tým má fungovat procedurálně. Nedefinuje jenom nějaké principy, ale vysloveně i ty procedury chování toho týmu, který zajišťuje v podstatě i tu zpětnou vazbu na konci každého sprintu, na konci větších dodávek. A to mi přijde, že možná to je ten důvod proč. Když to třeba porovná s tím Kanbanem, tak ten Kanban je v tomto volnější.

#### 7 Napadá Vás příklad jiné agilní metodiky, která je vhodná k implementaci?

Za mě jen zřekapituluju to, co jsem říkal. To byl ten SAFe, ten Scale Agile for Enterprise. Byl by to asi i ten LeSS, což teďka nevím přesně, jak ta zkratka je. To je prostě Scrum of Scrums. To je ta druhá. A ten Kanban. Potom vlastně, ony ty agilní metodiky mají řadu technik, které jsou specifické, pohodlné pro softwarový vývoj, tím, že jsem ze softwarového vývoje. Ať je to test-driven development nebo domain-driven design, behavior-driven design. To jsou pak konkrétně ty techniky agilní, ale není to ucelený rámec. My kromě toho řízení těch dodávek Scrumem, tak používáme i některé ty techniky pro softwarový vývoj specificky.

### **Problémy při implementaci**

#### 8 Byl/a jste zaměstnán/a ve firmě během implementace agilních metod?

Byl jsem součástí dodávek, kdy se ta metodika používala.

- a. Setkal/a jste se při implementaci agilu s nějakými problémy?
- b. Pokud ano, s jakými? Jak jste jim čelil/a?

Největší problém asi, který je systematický, je v tom začlenění takhle fungujícího týmu v podstatě do zbytku organizace. Aby ten tým byl schopen respektovat nějaké principy a pravidla, které tato organizace si potřebuje vynutit, ať už v oblasti bezpečnosti a architektury, finančního řízení. A vlastně napojit to vůbec na ty procesy zbylé ve firmě. Protože třeba to finanční řízení dodávek v týmu je trochu problematické. Tady my fungujeme jako organizace jakoby na bázi ročního plánu finančního. Prostě když skončí rok, tak ty peníze zmizí a zase

musí znovu dělat ten plán. Tady to fungování jde ruku v ruce s waterfallovým přístupem, kdy se na začátku opravdu udělá ta analýza toho celku a z toho se udělá finanční plán a jede se. A vlastně ty adaptivní metody... Kdy já mám fixní rozpočet, ale nemám ho daný na základě nějaké analýzy, end-to-end analýzy, tak u nás je potom problém takový rozpočet jakoby obhájit a komunikovat. To byl ten velký problém, jak tahle fungující tým namapovat na change management process ve firmě, která funguje na základě těchto principů. No a potom v rámci tohoto týmu jsou takové otázky, když se dělá třeba hodnocení náročnosti tasků. Tak jestli pomocí bodů, nebo pomocí man-dayů – co je vlastně lepší? Ale ukázalo se nám, když jsme dělali hodnocení těch tasků v tom sprintu i pomocí bodů...tak vlastně po dvou, třech sprintech jsme měli odhadnutou velocity toho týmu, vlastně kolik bodů on je schopný zpracovat, když se to udělá. A pak nám vlastně přestalo vadit, že to nejsou man-dayi, že to jsou jen body. Ale bylo to na začátku trochu jako problém, tak jsme to vnímali u této vlastní metodiky. Potom, co je důležité u tohoto fungování, aby ten tým seděl pohromadě, mohl spolu interagovat. Tato budova má své omezení kapacitní, a pro ty změnové aktivity mít možnost ad hoc udělat, sestavit tým...je to možné, jen když je volné místo. Když volné místo není, tak to vlastně nejde. To je taky omezení, na které jsme narazili. To tolik nesouvisí s metodikou. A...vlastně trochu jo. Ona s tím počítá, na to jsme narazili jako na problém...takže pak jsme museli dělat stand-upy tak, že ty lidé seděli mimo, nebyli pospolu...pak ale ty stand-upy byly, ale to bylo v takovém stísněném prostředí, a nebylo to úplně ono.

#### 9 Víte, nakolik je těžké, nebo snadné sestavit tým, který by vyhovoval agilním principům?

Pro mě je klíčem ta seniorita těch lidí. Ten senior jednak v tom, že v té dané oblasti ten člověk je dostatečně vyspělý v tom, že on je schopný dodávat to, co má dodávat. Protože v takovýhle firmě, jako je ta naše, je řada lidí, kteří jsou jakoby úředníci. Kteří jsou jakoby v nějaké oblasti a nejsou schopní produkovat reálné výstupy, spíš organizují tu práci. Takže to je takové první omezení. A pokud se sežene člověk, který má tu schopnost, tak on ještě musí mít schopnost jakoby k přesahu toho vnímání toho problému. Protože to je dost malý tým, který má entitu, entitu dodávky, takže ten člověk nesmí být třeba čistě jen vývojář, ale musí mít trochu architektonické myšlení, musí trochu rozumět tomu businessu, který se řeší. Takže to je druhý rozměr toho problému, té kvality člověka.

#### 10 Pokud někdo do týmu nezapadá (jsou v týmu interpersonální konflikty), jak to řešíte?

U nás se to nakonec řešilo tak, že v rámci dané dodávky nějakého toho prvního produktu, tak prostě ty lidé museli spolu nějak vyjít, protože nebylo možné udělat tu výměnu. Ale pak je to samozřejmě o tom, že ty agilní týmy vznikají pro nějakou konkrétní dodávku, a pak zanikají,

aspoň v tom našem prostředí zanikají. Když startuje něco jiného, tak zase vznikne ten tým a ta alokace do toho týmu probíhá v podstatě na základě té předchozí zkušenosti, takže tím probíhá ta eliminace lidí, kteří třeba nejdou dohromady, nebo neperformují, jak mají. Nemáme žádnou sofistikovanější metodu. [...]

## **Dodržování principů agilu a praxe**

11 Je u Vás používán princip kolektivní viny (za chyby zodpovídají všichni členové týmu)?

Je to super princip [smích]. Používáme ho i mimo agilní týmy [smích]. Takové to heslo „miluji týmovou práci, protože můžu schovat svou odpovědnost za odpovědnost kolektivu“. Z toho se u nás hodně těží.

12 Jak byste popsal/a vztah Vašeho týmu se zákazníkem?

Když se na to podívám v rámci fungování toho agilního týmu, tak tam je ten zákazník svým způsobem členem toho týmu. Tam je klíčová osoba toho Product Ownera. [...] Teď mi došlo, že naplnění role Product Ownera...že to musí být takový člověk, který je opravdu schopný za ten business prioritizovat ty fičury, které on chce do toho dostat, a musí mít tu pravomoc tu prioritizaci provést. Nesmí to být, že závisí na někom jiném, to pak hrozně komplikuje to fungování týmu, protože on není autonomní a není tak rychlý...protože se čeká, až se zase sejde někde nějaká skupinka lidí a ty lidé nemají čas a ta dodávka se tím prodlužuje. Tam vnímám, že vztah s tím zákazníkem je v případě dodávek Scrumem, třeba těmi adaptivními a agilními metodami lepší, protože ten zákazník dostane ne *to*, co se snažil popsat na začátku, ale dostane *to, co chce*, když je to *hotové*. Což je velký rozdíl. My jsme tady měli řadu projektů, které běžely déle jak rok. Samozřejmě během té doby se situace na trhu i uvnitř firmy změnila. A i když analýza byla dostatečně přesná, tak ale ten stav se změnil a dodalo se to, co se poptalo, ale už to nebylo vlastně potřeba, a to tam byla řada rozčarování. Těch rozčarování, nebo těch nepotkání se s tím zákazníkem, je samozřejmě mnohem více. Ta analýza není někdy tak přesná, aby se zmapovalo to vnímání zákazníka, který poptává tu změnu. Takže toto agilní metody určitě řeší.

13 Jak často se stává, že Product Owner nebo Scrum Master má pracovat s více než jedním týmem? Může to způsobit problémy v projektu?

Myslím si, že ten Product Owner...že by měl být spíš dedikovaný pro ten daný produkt. Ona ta činnost zabere hodně času a vyžaduje to celé jeho nasazení. U toho Scrum Mastera si myslím, že tam není problém, pokud jede přes dva až tři týmy. Když jsme tak teďka dělali „sizing“ nějakých třech, čtyř dodávek, právě jsme na to narazili. Protože nemáme všechny lidi interní,

takže jsme sháněli u externích firem, které mají mezinárodní zkušenost, a právě jsme s nimi řešili ten kapacitní „sizing“ tady toho, s tím Scrum Masterem. [...] Máme tak dva zkušené Scrum Mastery a zbytek musíme brát externě.

14 Vyžadujete od PO a SM nějaké certifikáty? Pokud ano, jaké?

Certifikáty ne, certifikáty nevyžadujeme. Spíš hledáme reference, kde už jinde se s tím setkal. Protože docela považuju za problém, a teď si ho tady možná i interně sami trochu vyrábíme... Když se snažíme udělat podstatnou změnu, transformační změnu...myslím, že použití agilních metodik *je* podstatná transformační změna, a snažíme se to vymyslet sami doma. S tím pochopením akorát jako pocitovým, neprožili jsme si nikdy, nikdy fail ty metody, abysme poznali, jak to celé funguje. Co do těch aktivit, které jsou pro nás klíčové, tak tam si bereme externí lidi, kteří jsou schopní dokladovat, že tuto roli už někde odehráli. A úplně ještě ideálně, že i někdy zažili nějaké selhání toho týmu a jsou schopní se z toho poučit. Takže nějaké certifikace...to vůbec neřešíme. [...] Člověk bez zkušeností by neměl šanci pro klíčové projekty.

15 Na kolika projektech může zároveň pracovat jeden člověk?

Rozhodně více jak na jednom. Nedokážeme si tady zařídit dedikovanou účast. Ale jsou to opravdu jednotky, třeba dva, tři projekty, kde jsou ti lidé schopní reálně fungovat. Tady s tím máme taky velký problém, protože ty role, které jsou pak na těch standartních změnových softwarových projektech, ať je to bussines analytics, solution architect a další, tak ty často mají na sobě 7–10 projektů a v tom reálně nejsou schopní pořádně fungovat. Projevuje se to na kvalitě jejich práce, ale v podstatě objem těch změn, který tady ta firma tlačí, je tak velký, že v podstatě tam se to množství lidí musí rozřezat na desetinky. [...] Agilní týmy jsou mnohem na to citlivější. Proto si to nemůžeme dovolit. Je to důležitý, aby ten člověk byl jakoby fokusován na práci, a tudíž maximálně dva, tři projekty, pro tu roli třeba toho vývojáře.

16 Popište prosím, jak probíhá komunikace mezi členy týmu.

Buďto kolegové sedí v jedné kanceláři, nebo používáme Slack.

17 Používáte nějaký software k plánování sprintů, meetingů, pro backlogy atd.? Pokud ano, uveďte název.

Pro agilní metody máme jednak Jiru, která má agilní moduly pro sebe. Máme Trello, kde se dělá vlastně plánování, a pak se to překlápí do Jiry. To jsou vlastně dva takové nástroje. Backlogy a meetingy jsou v Jire.

## 18 Jakou délku sprintu obvykle máte?

Dva, maximálně tři týdny.

## 19 Pokud v průběhu sprintu nějaký úkol není splněn, co se s úkolem dále stane? Jak často se Vám to stává?

Na konci sprintu se v tom review – co se udělalo, co se neudělalo – tak v té respektive řeší „proč“, jestli tam byly externí důvody. S tím, jak jsme provázaní, jak to není ten malý tým, který sám něco dodává s minimem vazeb na to okolí, se zkoumá, jestli to bylo skrze ty vazby, jestli jsme schopní to nějak ovlivnit, a pak se to parkuje. Buď se to hodí do následujícího sprintu, když to byla nějaká drobnost. Nebo když je to něco podstatného, tak se to parkuje bokem a pak se to řeší bokem toho týmu. Tým si spíš pak vyžádá roli vně týmu, která to pomůže řešit. Na to jsou právě dobrý ty větší metodiky, třeba ten Scale Agile for Enterprise, které na toto mají už instrumenty. [...] Já si myslím, že vždycky se nám stalo, že něco nebylo splněno. Nebo ne vždycky úplně, ale je to více jak 50 %, že se nám naskytl nějaký bod, který se nepodařilo splnit.

## 20 Jaká je střední délka plánovacích meetingů?

Když se dělal planning, tak hodina, hodina a půl přibližně. Ona je taky otázka, čím je ta doba ovlivněná. U nás je často ovlivněná tím, že jak mají lidé zaplněné kalendáře, tak není možné najít větší společný prostor, aby se sešli. Že ještě při tom planningu ty lidi nejsou jako nalokování úplně na tu danou aktivitu a je prostě problém najít další společný čas.

## 21 Daily stand-up by měl ideálně být každé ráno ve stejnou dobu, ve stejném místě, ve stoje a maximálně 15 minut. Jak to máte Vy?

Na řadě projektů to takhle funguje, takže se to daří dodržovat. I to, že to je ve stoje, to, jak jsem říkal, že je problém s místem... [smích]. Nemají šanci se tam jako uvolnit, sednout si a pasivně konzumovat ten meeting. Ten pevný čas je pro nás dobrý i v tom, že se to nechá zarezervovat na začátku lidem do kalendáře. Vybookujeme tím jejich čas. Samozřejmě jako zápasíme s tím. Tam by bylo dobré, aby byli všichni na tom stand-upu, aby si vyměnili informace. Nebo když jeden minul, tak by byli na tom druhém. U nás dochází k tomu sdílení lidí, takže jsou na dvou až třech dalších meetinzích a mají i jiné povinnosti a stává se nám, že ty lidi dvakrát až třikrát v týdně vypadnou z toho. Takže se to potom řeší přes nějaké jiné nástroje, anebo přes jiné kanály, vlastně skrz nějaké další nástroje. Když jsme zmiňovali ty nástroje, tak ono to nesouvisí s tím agilním řízením, ale spíše s vývojem. Používáme třeba Slack na komunikaci. Řeší, co bylo na stand-upu, takže tam se napíše, co bylo na stand-upu, a tím se jako úplně sdílí.

## Výsledky implementaci

22 Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily? Myslím si, že když potom porovnáš tu kvalitu výstupů s klasickým waterfallem, tak byla zkušenost vždycky lepší. S tou konkrétní dodávkou, tím přímým zákazníkem, tím přímým konzumentem. Je pravda, že třeba útvary, které reprezentují nějaké principy...jak jsem řekl, architektura, bezpečnost, finance, které bysme měli dodržovat, tak ti s tím ne vždy byly [útvary] spokojení. Tam se prostě obětuje ta compliance, třeba z bezpečnosti, za to, aby se doručilo tomu klientovi v podstatě ta funkčnost, třeba nějakým work roundem. Takže klient měl z toho lepší zážitek. Firma jako taková, ne vždycky a my doufáme, že pokud nasadíme ty komplexnější metodiky, takže nám pomůžou tenhle stav napravit.

23 Mělo zavedení agilu dopad na rozsah dokumentace k projektům?

Mělo. Mělo, mělo. Ty standardní projekty, které jedou tím waterfallem, měly systémovou specifikaci, dokument o 15 kapitolách. Když se vyplňoval, tak to bylo minimálně 20 stránek, kde člověk našel aspoň ty základy toho, co potřeboval. U agilních týmu ta dokumentace má jakoby nižší prioritu než ten vlastní kód, ten vlastní produkt. Takže se ta dokumentace reálně potom dělá v rámci toho Trella nebo tý Jiry, k tím jednotlivým taskům, které se řeší. Je to horší najít, pokud... Ten tým se tam v tom orientuje jako takový, ale problém je, že to má pak navázat někdo jiný, tak neví, kde tu dokumentaci najít, a podobně. Todlento nemáme moc vyřešené a je to jeden z tablů.

24 Bylo podle Vašeho názoru zavedení agilu přínosným pro Váš podnik?

Prozatím myslím, že ne. Že ten rozměr těch agilních metod...že kdybych to vzal procentuálně, tak je 5–10 % z těch změn se dělá těma agilními týmy. Že jako celá firma, to ne. Ale jakoby už ale jako firma jsme pocítili v jednotlivých dodávkách krásu, nebo tu výhodu agilní metody. Máme dvě aplikace tímhle způsobem doručené a tam se v podstatě právě podařilo tomu zákazníkovi doručit to, co on na začátku nevěděl, že chce. A v tom bylo to kouzlo.

## Hodnocení a doporučení

25 Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.

Přišlo mi tedy, že ten největší pozitivní dopad je v tom, že ten zákazník si může v průběhu implementace jakoby iterativně upřesnit, jaká je jeho vlastní vize, nebo spíš konkrétní představa. On nějakou vizi na začátku má, ale nemá konkrétní představu, ještě navíc

otestovanou tím konzumentem té služby. My vlastně v rámci toho iterativního vývoje řešíme projekty, které jsou o těch koncových aplikacích, které nějak interagují, buďto se zákazníkem XY nebo s tím interním zákazníkem, to znamená s poradcem na pobočce. Tam je hodně důležitá ergonomie toho procesu v té aplikaci, v tom UI. Takže my tam v rámci těchto projektů zařazujeme UI/UX testy, děláme testování a v rámci toho testování vypadává feedback toho typu, že někdy jsme nuceni vlastně zahodit celý ten UI a udělat to úplně jinak. Má to dopad i do dat, do datových entit a podobně. Tohle si ten zadavatel vlastně na začátku jakoby vůbec neuvědomuje. A to je ta ohromná výhoda toho iterativního procesu.

26 Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?

Asi jsem o nich mluvil. Ten negativní dopad... Je to ten alignment s tou větší strukturou, s tou celou firmou. Neschopnost dané firmy ve větším měřítku využít týmy, protože nemáme ty lidi k tomu. A co je zajímavé, když tady vlastně mluvím o tom, že ta metodika pro celou enterprise nás zachrání, tak ona nás vlastně taky nezachrání. Podle těch informací, co máme třeba od ING, kdy vlastně udělali tu velkou změnu, ta změna byla podpořena metodikou pro celou firmu, tak i oni vlastně říkají, že: „Po 2 letech používání máme problém s tím, že máme ve firmě tak trošku anarchii. Firmu nedokážeme řídit, protože nemáme informace. Lidé nesprávně pochopili tu volnost, kterou dostali.“ To byl jakoby jeden problém. A druhý problém byl, že ty lidi se najednou ocitli v nových rolích, nedokázali si tam najít svoje místo a byli demotivovaní. Začali i schopní lidé odcházet díky té demotivaci. Takže si myslím, že nestačí jenom ta správná metodika, dostatečně robustní, ale je to opravdu transformační změna. Lidé musí pochopit tu změnu. Je to těžký a dlouhotrvající proces. Takže podle mě to nejde nasadit jako mávnutím proutku, ale musí se to v té firmě budovat iterativně, postupně rozšiřovat. Znáš takový termín „krystalizační jádra“. Musí v podstatě shrnout krystalizační jádra v těch lidech, co mají tu zkušenost a jsou tu zkušenost schopní předávat dál.

27 Co byste doporučoval/a lidem, kteří chtějí dané metodiky implementovat?

Aby si ty lidi vlastně snažili uvědomit, proč tu metodiku vůbec nasazují. Jestli opravdu se jim bude řešit ty problémy, které potřebují vyřešit. A jestli jim nebude vadit, že si tam zavlečou problémy, které s tou metodikou přichází. Už jak jsem zmiňoval ty problémy z ING. Ty problémy, které se tam zanesou, jsou asi řešitelné, ale je otázka, jestli ten celkový výsledek potom ve firmě bude lepší, nebo nebude. Ale když to třeba vezmu na situaci v bankovníctví, které se hodně opírá o IT, tak si myslím, že tato metoda je teďka vhodná k nasazení. Ten důvod je ten, že máme relativně už komoditizované ty bankovní služby. Žádná banka už teď nevymyslí

nový finanční produkt sám o sobě. Ty produkty se od sebe mezi těmi bankami už moc neliší, takže se nemusí dělat nějaké zásadní změny v těch systémech vevnitř, které dělají ty platební funkce. Tam, kde se snažíme vzájemně odlišit, je, jak ty produkty nabízíme. V jakém kontextu, v jakém digitálním kontextu, na jakém digitálním touchpointu a tam se odehrává většina těch změn. Pro toto jsou ty agilní metodiky vhodné. Protože už neimpactují velké systémy, nechá se tam aplikovat takový ten lean product development. Prostě všechno to jde ruku v ruce. Mám už komoditizované IT služby, nemusím vymýšlet něco nového, takže nemusím mimo ty sprinty startovat nějaký dlouhotrvající task na postavení úplně nějaké nové technologie. Všechno se tak jako hezky potkává teď v tom bankovním světě. Tam se to třeba vyplatí. Ale jsou odvětví, asi průmyslové, kde to takhle není a kde by to třeba nedávalo smysl. A i my tady máme vevnitř pár aktivit, které takovým způsobem nejdou dělat. Je to třeba konsolidace našich bankovních backendů. To jsou právě ty systémy, které realizují ty základní platební funkce. To jsou dost komplexní systémy, jsou právě o těch finančních operacích a ty vlastně nejde měnit takovýmhle iterativním způsobem. Tam není žádný mezikrok možný. V podstatě buďto ten zůstatek sedí nebo nesedí. Tam ta agilní metoda nejde.

## 28 Chybělo Vám něco v našem dotazníku? Chtěl/a byste jej nějak doplnit?

Teď si musím v hlavě projít, o čem jsme vlastně mluvili a co mě tak napadalo, když jsem přemýšlel, o čem bysme mohli mluvit. Ale my jsme se vlastně dotkli všech těch aspektů. Vhodnosti, nevhodnosti, co nám na tom vadí, lidí. Myslím, že otázky máte kvalitně poskládané. Střílí vlastně po celém tom portfoliu toho změnového řízení. Za mě je to prostě pohled člověka, který pracuje vlastně v IT v bance, takže se to hodně motá kolem software development lifecycle a product developmentu, jako finančních produktů. A jak už jsem říkal, tato situace nahrává u nás používání těchto agilních metod, takže to podle mě zaregistrujete v tisku, že se tím banky začnou chlubit, že nasadily. Protože i to ING se tím relativně chlubí. V MF Dnes se o tom nepíše, ale třeba to ING v těch odbornějších časopisech. Tam se tím hodně chlubí. Protože dneska vlastně ta schopnost rychle inovovat se hrozně cení a když někdo přijde s nějakou změnou schopnosti, aby rychleji inovoval, tak se tím určitě pochlubí. Můžu se nad tím potom ještě v klidu zamyslet, kdybyste chtěli, protože myslím, že mám ten dotazník k dispozici. [...] Klidně mi ještě pošlete aktuální verzi a já na to mrknu, jestli bych vám k tomu nedal nějaký input.



## **Příloha 3. Přepis rozhovoru s respondentem č. 2**

### **Obecné (úvod)**

#### **1. Připadají Vám agilní metody přínosné, nebo ne?**

Rozhodně ano. Já jsem opravdu velký fanda agilních metodik, protože mám víc než 30letou zkušenost se softwarovým vývojem. Vlastně jsem začínal na akademické půdě, na Ústavu informatiky, s neuronovými sítí. Vlastně jako vývojový tým, pracoval jsem řádu let v takových velkých společnostech jako REUTERS, Sun Microsystems a vlastně NN IT Services. A ve všech těch firmách jsme v podstatě aplikovali aspoň nějaké prvky agilní metodiky. Ze začátku jsme tomu ani neříkali „agilní metodika“, ale to prostě principiálně vedlo k těm základním prvkům těch metodik, protože jsme pracovali většinou v malých týmech, a ty požadavky se vyvíjely během toho času, takže klasická waterfall metoda těžko fungovala. [...] Ten vývoj agilních metodik byl těsně spjatý s mým životem. V době, kdy agilní metody ještě nebyly ani příliš známy, jsme pracovali v malém týmu a pak jsme se seznámili s extrémním programováním. Principy jako kontinuální integrace nebo test-driven přístup jednoznačně byly věci, které mě hodně ovlivnily. Pak jsem se seznámil se Scrumem, který se vlastně vyvíjel z těch původních metodik. Na Scrumu mi právě zahrálo to, že je obecný. Že ty metodiky, jako je extreme programming, tak vlastně jsou spjaty a vycházejí ze zkušeností softwarového vývoje. Ale Scrum funguje obecně. Já jsem třeba stavil svůj dům na Scrumu. Víím, že třeba americká armáda používá Scrum, i ve vážných operacích, jo. Nebo i marketingové kampaně se dají dělat ve Scrumu. Prostě je to obecný. Je to project management framework, na rozdíl od třeba právě test-driven development.

#### **2. Proč je dnes důležité se zabývat agilními metodami?**

Já si myslím, že rozhodně ano [smích]. Já si myslím, že pro společnosti, které vyvíjí softwarový produkt, tak je to dneska základní metoda vývoje, kterou když neadoptují, tak se budou těžko přizpůsobovat tomu trhu nebo těm měnícím se požadavkům na tom trhu. Podle mě, je to otázka přežití. Firmy, které prostě zůstanou strnule používat waterfall metodiky, nemají šanci rychle se vyvíjet s tím množstvím technologií, kde ten business se také velmi prudce mění. Jak se mění prudce požadavky a technologie, tak tam ty klasické metody waterafallu, kde vlastně předstíráte, že znáte budoucnost, tak to moc nefunguje, no. Ale dneska si myslím, že Scrum už je v podstatě klasika. Scrum se sám jako takový vyvíjí a napojuje se na produkci. Propojení DevOps s tím supportem a s operations třeba.

### 3. Jaké výhody a nevýhody mají podle Vašeho názoru agilní metodiky v porovnání s waterfallem?

Já si myslím, že ty agilní metodiky mají opravdu schopnost pomoci dodávat produkty tak, jak si ten zákazník opravdu přeje. A je to bezpečnější, pokud se to aplikuje správně. Myslím si, že je tam možnost reagovat právě na změnu těch požadavků. To, o čem jsme teď kon po práci bavili, je to, že zákazníci často přesně ani nevědí, co chtějí. Teprve když vidí první mock-upy, screenshoty, prototypy a mají možnost se k tomu pravidelně vyjadřovat, tak to je právě velká výhoda právě toho Scrumu. Myslím si, že systém toho Scrumu je velmi dobře nadefinovaný. Minimální sada pravidel, které umožňují, když se dodržují, vlastně velmi efektivně dodávat ty požadavky. Na rozdíl třeba od waterfallu nebo nějakých tradičních metod, které jsou velmi složité a z hlediska toho procesu jsou těžkopádné a těžko se implementují, tak ten Scrum podle mě je možné nasadit celkem rychle. Myslím si, že je to o rychlosti vývoje a o kvalitě. Nemyslím „rychlost vývoje“ jako kolik řádků, kódu za minutu, ale rychlost odezvy na nějaký požadavek. [...] Já myslím si, že jsou situace, kdy waterfall je lepší nebo je efektivnější. To je v situacích, kdy vlastně je ta technologie velmi dobře známá, ten tým v té technologii pracuje už delší dobu, ty požadavky jsou naprosto zřejmé a ten tým dokáže vlastně velmi dobře předpovídat. Pokud jsou třeba nějaké překladatelské služby nebo prostě nějaké transformace, kdy ten tým má nějaký vstup, třeba nějaké požadavky v nějakých tabulkách... To už nepovažuji skutečně za softwarový vývoj, ale je to něco, co je rutinní a co se dá dál skutečně dobře předvídat, jak dlouho to trvá. Pak je možné docela dobře vytvořit ten waterfall a ta predikovatelnost je tam daleko vyšší. Ale upřímně si myslím, že to samé se dá udělat v tom Scrumu. A nemyslím se, že to znamená, že o moc vyšší overhead v tom waterfallu. Možná nevýhoda taky je v tom, že vyžaduje jako větší zapojení těch lidí v týmu, že...nevím, jestli je to výhoda nebo nevýhoda, ale každopádně ne všichni a ne všechny firmy mají tu kulturu nebo firmu nastavenou tak, že by byli schopní se přizpůsobit. Třeba pokud ta firma má organizační strukturu tak, že business je zvyklý dodávat waterfallem, takže když mu vnucujete Scrum, tak řekne, že ho to nezajímá, mám ty business požadavky na deseti řádkách a za půlroku přijďte, já chci váš produkt, jo. A v takových situacích je to trochu složitější, no. Ale tam podle mého waterfall také nepomůže [smích]. [...] Já si myslím, že ta komunikace v tom týmu je naprosto klíčová. Proto taky není možné ty týmy nafukovat víc než třeba osm, devět lidí. Když je to přes 10 lidí, tak už je to trochu problém, protože podle to v těch komunikacích mezi členy toho týmu vyrůstá exponenciálně. Každý metr se počítá, jo. Aspoň to je moje zkušenost [smích]. Myslím si, že to je otázka sdílného zážitku. Toho Scrumu, že jo. Ten tým, když má tu informaci, nebo vnímá ty požadavky jako tým, což znamená, že je stále spolu a tu informaci integruje do sebe. Tak je

skutečně schopen pracovat jako jeden organizmus, zatímco když každý člověk pracuje zvlášť a předává sety, části, tak je tam velký overhead a zdroj nedorozumění. Když máte třeba system requirement specifikaci a předáte někomu dalšímu, z toho udělá design a udělá další implementaci. Tím, že třeba ti lidé nejsou do toho zapojení, v tom waterfallu, tak často neví, jak s tím být, a jsou tam zdroje i různých flámů. Princip „tichá pošta“ se tomu říká. V tom waterfallu se často ta informace, kterou váš zákazník skutečně potřebuje, má tendenci ztrácet [smích]. Nebo není úplně přesná, no. Speciálně když třeba se firma rozhodne aplikovat taky waterfall s outsourcingem, že třeba ten tester testování udělá v Indii. To je úžasná věc. To je trošku ironické, protože moje zkušenost opravdu byla velmi tristní. Myslím, že to je otázka nejen kultury, ale i vlastně očekávání. Že třeba ty týmy v té Asii se stydí ukázat, že tam jsou chyby [smích]. Takže vám posílají excel, který je celý zelený, a pak to zkusíte a to nefunguje [smích]. Ale to jsou prostě věci, ze kterých se dá poučit. Měl jsem třebamožnost zavádět Scrum v Bangaloru, v Indii třeba. Takže...jako pracuju s týmy v Malajsii... Té kulturní rozdíly jsou důležité pro to, jak nasadit třeba Scrum, ale to neznamena, že to nejde. Rozhodně to jde, ale třeba jste špatně porozuměli té mentalitě. A skutečně lidi do toho zatáhnout. V okamžiku, kdy jsou jakékoliv bariéry, tak tím je to horší. Nevím, jestli to odpovídá...[smích].

## Scrum

### 4. Máte zkušenosti s metodou Scrum?

[...] 30 let mám se softwarovým vývojem vlastně. Školu jsem skončil těsně před revolucí. Vlastně ještě ve škole jsem vyvíjel nějaký software pro analýzu, aktivity mozku, brain mapping a tak, ale to byl v podstatě izolovaný vývoj. V té akademii to už byl pětičlenný tým, který se rozrostl v desetičlenný tým, a tam jsme už prostě museli rozdělit ty týmy na dva. To byl moment...my jsme ani nevěděli, že existuje nějaký Scrum. Ale to bylo asi před 20 lety, no. Nebo i ještě dýl. Ale my jsme věděli, že když ten tým roste, tak to už prostě nešlo...nešlo tu komunikaci v rámci týmu řídit, tak jsme přirozeně rozdělili. Takže to je taková zkušenost, kdy mi vlastně ukazovala...pak mi lidí upozornili: „Hele, věděli jste, že existuje etreme programming?“ A se Scrumem jako takovým mám zkušenosti 15 let. Vlastně, když jsem pracoval v Sunu...Sun Microsystems. Nejdřív jsme tam nasazovali Extreme programming, pak vlastně Scrum. [...] Kanban je rozhodně věc, kterou se dá kombinovat. Myslím si, že obecně i Scrum vychází z těch lean metodik. To je vlastně z automobilního průmyslu, ty principy, to asi znáte líp než já. Mně se třeba líbí kombinace Scrumu a Kanbanu. Tomu se říká Scrumban. A myslím si, že je to dobré, protože to drží ten tým trochu na uzdě. Protože někdy ty vývojářské týmy mají tendenci rozpracovávat příliš mnoho stories paralelně. Pak se to těžko kočíruje. A ten

Kanban vám vlastně v rámci toho proudu těch požadavků umožňuje monitorovat množství těch rozpracovaných tasků. Je to vhodné třeba i pro supportové projekty. Viděl jsem třeba používat Kanban i na projektech, které jsou čistě maintenance třeba. Někaký software, který měl stream nějakých defektů nebo prostě nějakých drobných požadavků. A ten produkt byl v aktivním vývoje, což znamená nějaké velké architektonické změny. A vlastně stream těch malých změn, kde ten Scrum neměl až tak moc velký smysl, tak tam ten Kanban mohl fungovat jako brouční linka. Ale možná právě to priorizování, regulování toho toku těch požadavků v rámci toho týmu no.

5. Uveďte prosím základní odlišení Scrumu od jiných agilních metodik.

To jsem už zmínil. Scrum je moc obecný. A že je to project management practice, nebo jak metodologie, která vlastně není pevně svázaná s tím softwarovým vývojem a vystihuje podle mého ten základní princip. Základní principy, na kterých třeba agilní vývoj pracuje.

6. Čím je možné vysvětlit takovou popularitu Scrumu?

Myslím, že to má dva aspekty. První, že to funguje [smích]. Když vývojáři vidí, že něco funguje, tak to používají a myslím si, že je to jednoduchá metodika. To znamená, že každý to pochopí. Ty některé termíny jsou tam trochu „funky“ a vypadá to z pohledu nějakých enterprise metodologií trochu podivně. Jak se říká „sprint retrospective“ nebo „grooming sessions“, tak někteří vice-presidenti se z toho otřesou „co tam potkáváte, ti, kteří to chtějí dělat?“. Na druhou stranu, když to člověk pochopí, tak myslím si, že je to pro ně snadno aplikovatelně. A myslím, že to má i sociální aspekt. Protože když máte ty lidi v těch týmech, v těch boxech, tak po určité době dochází k vyhoření. Ten tester, když pořad jenom testuje, tak se z toho prostě může zbláznit. Když ten člověk chvíli něco programuje a něco testuje, vydělává nějakou další roli, tak v tom týmu můžou střídat. Máte tam třeba principy párového programování. Tak v ten okamžik ta sociální jednotka toho týmu funguje daleko lépe. Tam ten týmový duch je znát a to dělá hrozně moc. Já myslím, že to, že je tam ta viditelnost, každý den se všichni sejdou a si říkají, co udělali, co dělají a co plánují udělat, a najednou se to stává velmi transparentní a lidé vidí a mohou daleko víc pomáhat. A naopak je těžko něco skryt. To je prostě na jeden okamžik transparentní. Samozřejmě ten přechod, nebo ten mainshift, je někdy těžkej. Speciálně třeba v nějakých týmech, které dlouhou dobu vyvíjeli v tom waterfallu, tak tam ten přechod není jednoduchý. Když máte nějakou organizační strukturu, která je velmi hierarchická, tak tam najednou to nasazovat je trochu problém. Ale jde to.

7. Napadá Vás příklad jiné agilní metodiky, která je vhodná k implementaci?

O tom jsem už řekl. Ty metody extreme programování včetně těch practices jako kontinuální integrace nebo párové programování. Nebo test-driven přístup. To jsou věci, které jsou přirozené doplňky ke Scrumu. Po případě, že děláte softwarový vývoj. Prvky jako Kanban, speciálně sledování těch tasků – to si myslím, že je velmi zdravá věc. Pokud prvky té lean metodologie, si myslím, se také dají používat. [...] Já si myslím, že to je právě takový trend, že pokud ten tým pochopí podstatu toho Scrumu, tak vlastně naráží na problémy jejích řešení. Jsou právě ty best practices, jak jsem říkal. Když nemají vytvořený automatický systém integrace nebo kontinuálního dodávání do nějakých testovacích prostředí nebo automatické testy, tak ten tým vlastně trochu trpí. Protože když každý sprint musí dělat nějaké vlastní testy, tak je to trápí, takže přirozený laik použije automatické testování třeba. Nebo pokud chtějí, aby měli feedback od toho zákazníka častější, tak samozřejmě, když mají automatický deployment toho testovacího prostředí, a ten zákazník má k tomu prostředí přístup, tak je to mnohem lepší. A pokud mají nějaké prvky jako code review nebo párové programování, tak se ta kvalita kódu rapidně zlepšuje. Když máte v týmu starší a mladší lidi, tak tam ten extreme programming spolu se Scrumem funguje tak, že vlastně pomáhají tomu vývojáři se velmi rychle učit, protože mají třeba staršího kolegu, a zase ten starší kolega nemá nějaké nápady, třeba nepíše tak rychle nějaký kód. A tím se vzájemně obohacují.

### **Problémy při implementaci**

8. Byl/a jste zaměstnán/a ve firmě během implementace agilních metod?

a. Setkal/a jste se při implementaci agilu s nějakými problémy?

Byl jsem v několika [smích]. Když třeba mluvím o Scrumu, tak to skutečně začalo v Sunu a tam jsem byl pět let a pak jsem se tím masivně zabýval v rámci NN IT Services, kde pracuju už 12 let teďkon. Čelili jsme obrovskému množství problém [smích], takže to je velmi složitá otázka. Kdybych měl vyjmenovat nějaké největší problémy... Myslím, že jeden z problémů je trochu neporozumění tomu, co je Scrum. Protože často někdo zaslechl „Scrum není žádná dokumentace“ nebo „Scrum je něco, co se dělá strašně jednoduše a nepotřebujeme žádnou architekturu“ [smích]. Nebo ten business říká „Tak to je, otevřené dveře do zadávání požadavků během vývoje i během toho sprintu“. A to jsou prostě nedorozumění, které potom vedou k velkým problémům. Já jsem zažil situaci na projektech, třeba projekt v Malajsii, kdy vlastně ten Produkt Owner byl na úrovni nějakého ředitele a ten vývojářský tým se tak ho strašně bál, že dělal úplně všechno, co řekl, jo. A to i když ten člověk řekl něco uprostřed toho sprintu, takže oni to adoptovali do toho sprintu a pak jejich backlog stále rostl. A on jako Product Owner nehrál tu roli Product Ownera, protože nevyvažoval ty požadavky, což znamenalo, že když něco

přibýlo, tak nic neubral. Takže se jednou ten projekt dostal do situaci, kdy vlastně to bylo neudržitelné. A běžel po střeše zásadní zásah jako. Někdy to jsou i organizační záležitosti, třeba spolupráce s tím businesssem vyžaduje ještě nějakého mediátora. Proto se v tom „novějším“ Scrumu rozlišují role Product Ownera a Business Ownera. Že třeba máte řádu stakeholderů, ale je ještě důležitý z těch stakeholderů ten Business Owner. A pak ten Product Owner sbírá požadavky nejen z toho businessu, ale i třeba nefunkční požadavky ze suportového týmu nebo z architektury, security, operations. To jsou prostě věci, které třeba ten Business Owner nevnímá. Kdyby jenom byl ten Business Owner Product Ownerem, tak vlastně by neprioritizoval ty požadavky, které jsou technické, stejně. A to je potom ta role Product Ownera, aby to vyvažoval, aby skutečně tu kvalitu produktu dosáhnul. I za tu cenu, že některé ty business požadavky bude moderovat nebo každopádně musí pracovat s tím businesssem na tom, aby kvalita toho produktu byla taková, jako si zákazník představuje. Včetně supportu, nebo e-security, kterou zákazník často bere jako samozřejmost.

b. Pokud ano, s jakými? Jak jste jim čelil/a?

[...] Jako...složitá věc. Jedna věc je, že ta osvěta je velice důležitá. My jsme třeba v ITS, které mají opravdu jako tisíc lidí v IT, v tom vývoje, tak jsme investovali hodně do toho vzdělávání. Ale samozřejmě nemůžete každého vývojáře takhle vyškolit, na certified Scrum Mastera, certified Product Ownera, je to drahý. Ale toto jsme udělali, že jsme se snažili tvářit jako takové ohnisko, jo, kde v každé oblasti, nebo doméně, jsme se snažili získat aspoň několik lidí, kteří měli zájem, kteří byli takoví „early adopters“. Do nich jsme investovali a oni se potom stali těmi trenéry Scrumu v těch jejich oblastí, takže je to jedna věc. Ale otázka je i podpora managementu. My jsme třeba měli situaci, kdy v té Asii ti vývojáře říkali: „Musíme vyvíjet ve waterfallu, protože zákazník to chce“, a vždycky analyzovali nějaké požadavky několik měsíců a pak něco vyvíjeli třeba dalších několika měsíců. Já jsem se ptal: „Jak to víte, že zákazník to tak chce?“ A on říkal: „My jsme se neptali, my to tak myslíme“ [smích]. Pak jsme jeli do Singapuru a opravdu se toho zákazníka ptali, že jsme vysvětlili mu, co to je Scrum, a on říká: „No to je přesně to, co my chceme“, jo. Takže často je to nedorozumění, rozumíte? „Já nevím, proč to nám dáváte ve waterfallu? My to chceme takhle!“ [smích]. Takže neporozumění, jo. Často to jsou i nějaké kulturní věci. V Indii byl problém, že třeba ten tým najednou dodával pomalejc, ale když jeden z členů toho týmu odešel na dovolenou, tak dodávali rychlejc. Byla to taková záhada a pak jsme zjistili, že vývojář byl velmi slabý, ale byl z vyšší kasty. A ti ostatní se mu snažili pomáhat a tím se vlastně rozptylovali [smích]. A dělali chyby a moc to stejně nepomohlo [smích]. Takže to jsou takové situace, kde je nutné zasáhnout prudce, třeba vyloučit člověka z toho týmu, nebo třeba dohodnout se na něčem. To neznamena,

že všichni členy toho týmu musí být super programátoři. To není pravda. Ale je důležité, aby ten tým byl schopný dodávat, aby tam byly ty role, ty schopnosti, které jakoby do sebe zapadají. Jsou schopní jak toho návrhu, tak i vlastně toho vývoje i toho ověřování té nefunkčnosti no. Myslím, že tam je hodně důležitá ta otevřená mysl. Myslím si, že to je strašně důležitý okamžik, kdy člověk uvažuje: „Jsem vždycky jenom tester, co tady ode mě chcete“. Tak jako je tam pak to těžký, no. Nebo: „Já jsem programátor, já bych třeba testoval,“ no.

9. Víte, nakolik je těžké, nebo snadné sestavit tým, který by vyhovoval agilním principům?

Já si nemyslím, že je to strašně těžký. Já si myslím, že pokud máte skupinu lidí, kteří mají vůli a chuť prostě spolupracovat, tak mají i ty profily, pro které je možné dodávat. Jak ten tým je možné poskládat. Samozřejmě tam hraje roli i nějaká chemie. Někdy jeden člověk říká: „Já s tímhle tím člověkem nikdy nechci pracovat“ a pak je to těžký no. Ale to se dá taky řešit. Když máte víc těch scrumových týmů, tak můžete je třeba proměňovat. Pokud špatně funguje v jednom týmu, tak je čeká druhý tým. Ale když člověk není komparabilní s žádným týmem, tak je to opravdu problém ho vnímat [smích], což se řeší jako složitějc. Zažil jsem situaci, když ten tým vlastně to rozpoznával okamžitě a upozorňoval na to velmi rychle. Ten tým je daleko citlivější a „diagnostický“ na takové problémy než třeba v nějakém waterfallu. Tam se řada lidí může jako by ztratit. Já jsem třeba byl na nějakém projektu v Paříži a tam bylo 50 vývojářů z Indie a oni prostě několik týdnů tam seděli a nic nedělali, když jsem tam přišel [smích]. A zjistil jsem, že oni čekají na požadavky prostě [smích]. Ale nikoho to nenapadlo. Nebylo to prostě zachycený, takže to je i chyba toho waterfallu, projektový management tam byl problémový. A jsou jako i daleko složitější situaci, takový extrémní management [smích].

10. Pokud někdo do týmu nezapadá (jsou v týmu interpersonální konflikty), jak to řešíte?

Já si myslím, že tam je strašně důležitá role toho Scrum Mastera. Scrum Master je něco jako máma. A ten Product Owner je něco jako táta. Aspoň to v tom evropském kulturním pojmu...[smích]. Ten Scrum Master chrání ten tým a zajišťuje mu takové prostředí, aby tam mohl dobře fungovat. Ale ten Product Owner říká, co jako dělat dál, jo, jakoby racionálně. A v případě, že tam dojde k nějakému nesváru, tak ten Scrum Master...nebo jako kouč před fotbalovým týmem...prostě jako když se hádají nějaké hráči, tak vystoupí ten trenér a řekne: „A teď si to prostě vyříkáme“. A sejde se s ním nějak stranou a každý předložíte své problémy a snaží se to řešit...techniky manažerské pro řešení jako problémů...[smích]. To asi nebudeme rozebírat. [...].

## **Dodržování principů agilu a praxe**

## 11. Je u Vás používán princip kolektivní viny (za chyby zodpovídají všichni členové týmu)?

Já nesnáším ten pojem viny. Každopádně si myslím, že je důležité, aby ten tým cítil zodpovědnost za ten produkt. A skutečně se domnívali, že je to jejich. To znamená, že pokud se k něčemu zavazují, pokud se o tom rozhodují...to by mělo být týmové rozhodnutí, že jo. V případě, že někdo s něčím jakoby nesouhlasí, tak pokud je argument toho týmu dostatečně silný, tak by měl prostě říct: „Dobře, nesouhlasím, ale udělám to“, a přesto přijme to jako kolektivní rozhodnutí toho týmu. Já jsem prostě nezažil takovou situaci, kdyby se ten tým úplně nebyl schopný dohodnout, jo. Ale myslím si, že kolektivní vina nebo prostě stresování týmu nevede k vyšší efektivitě. Jsou třeba různé školy Scrumu, jo [smích]. Jsou názory, že by ten tým měl skutečně natvrdo zavázat, že dodá ty stories nebo že dodá tolik story pointů během toho sprintu, a když nedodá, tak je prostě šikanují nebo potom ten tým trpí pocitem viny, že to nedodal nebo tak. Myslím si, že je to špatně. Myslím si, že je důležitější, aby ten tým prostě fungoval v nějakém rytmu, aby měl nějaký dobrý pocit z toho, co dělá, a když se něco nestihne, tak se prostě nestihne, jo. Ale důležité je, aby ten mechanismus fungoval, což znamená, že před sprintem všechny stories jsou ready. Že ten tým může opravdu jet a skutečně vyvíjet, jo. Že během toho sprintu je nikdo nevyrušuje a můžou se skutečně soustředit na tu práci. Že ten Product Owner v případě jakékoliv nejasnosti je tam k dispozici a je schopný pomáhat. To je často situace, když Product Owner je vzdálený nebo třeba jenom part-time, tak to může být zdroj problémů, protože ten tým bude pak vytvářet své nějaké předpoklady, a nebudou pravdivé a to se zjistí až pozdějc. Takže prostě na to dávat pozor. [...] Já si myslím, že by ten tým měl vnímat ty problémy, jo. A že problém jednoho je problém všech, to si myslím, že to je. Ale nemělo by se v tom týmu jakoby šířit ten pocit viny. Protože když ten tým je ve stresu, tak dělá víc chyb. Já jsem třeba zažil takovou věc, že můj učitel Scrumu, americký Scrum Master, tak byl v americké armádě a dalších velkých firmách, a říkal, že nezáleží na tom, kolik těch story pointů ti lidé udělají, ale když ten tým řekne: „Tohle si myslíme, že to uděláme“, a mají jasně dané priority, a pak se snaží to udělat, tak se ten tým jakoby vyvíjí, a ze začátku přesnost toho odhadování je daleko menší. Třeba ty týmy mají tendenci přeceňovat své schopnosti, ale pak se to zklidní. Pak ten tým se formuje, to znamená, že velocity neboli ta rychlost je nižší, i třeba malinko nižší, než u toho waterfallu, ale pak se to velmi rychle vrátí. Dělali jsme statistiky na velkých vzorcích a tam to bylo hezké vidět, že ty týmy mohou se zrychlit. Samozřejmě je to problematické napříč týmama, počítat story pointy, protože každý jiný tým to má trochu jinak nadefinované, tu velikost story pointů, a nemusí to být stejně projekt od projektů a tak. Takže je to trochu zrádné a může být o špatném chování týmu. Pokud například management rozhodne, že chce zvýšit rychlost vývoje s tím, že nastaví tím lidem



KPI, něco jako „zvýšíme počet story pointů o 30 %“, tak to samozřejmě začne tím, že ti lidé začnou nafukovat ty stories [smích], protože tím zajistí ten bonus, ale to nepomůže ve Scrumu. [...] Já si myslím, že ten princip jakoby je u nás používán, ale neřekl bych, že to je vina. Já si myslím, že pokud je tam problém u jednoho, tak to je problém celého týmu. V tomto smyslu ano. Ale to, že by se ten tým neustále vinil, tak to si myslím, že není dobrá věc. Já si myslím, že to pomáhá v těch retrospektivách, takové to věci rozkrývat a řešit.

## 12. Jak byste popsal/a vztah Vašeho týmu se zákazníkem?

To je složitá otázka. Když vezmu například ty týmy, které tady máme v rámci NN IT Services, tak jako...já jsem teď třeba nějakým Product Ownerem v nějakém jednom projektu, jako konkrétní příklad...tak vlastně ten zákazník je v rámci firmy. To znamená, že my nemáme zákazníky jako mimo, ale to bude další divize, rozumíte? Takže je to vlastně...NN IT Services je IT firma, obrovská, tady například v Praze máme přes tisíc zaměstnanců...a ta firma zajišťuje IT služby NN Businessu. To jsou ty letadla...lodě [smích], ty nakladáky a tak. A ten business samozřejmě má nějaké procesy, které jsou podporovány nějakým systémem, a ty požadavky na systém zformulují lidé, kterým říkáme prostě Business IT. Takové oddělení, které sbírá ty požadavky z toho businessu. A my budujeme něco jako „supply chain“ [smích] požadavků, což znamená různé požadavky z různých částí světa. Ty požadavky se postupně nějakým způsobem filtrují. A to dodávání vlastně části tohoto řešení není jenom vždycky tou naší firmou ITS, ale i řadou vendorů, se kterými spolupracujeme, takže některé ty firmy dodávají komponenty, třebametry, adoptivní učení, nebo nějaké roboty atd. Třeba jsou ti, které nevyvíjí, ale spolupracují s velkými firmama, dodávají nám komponenty, a my to integrujeme. Takže je tam i složitější prvek no. A taková ta „vrstevnost“ je trochu brzdou. Protože poměrně často ti vývojáři nemají přímý kontakt s tím businesssem. Takže jeden z těch způsobů, jak to řešíme, že třeba posíláme ty programátory do nějakého hubu. Třeba v Lipsku je evropský hub, podívají se na ta letadla a jak to tam běží. Nebo třeba do nějakých skladů. Já jsem třeba byl ve Španělsku při nějakých testech na tom automatizovaném skladu, jo. A najednou řádky toho kódu začínají mít jiný význam. Člověk si uvědomuje, že když například v Šanghaji je chyba v tom kódu, obrovská hromada balíčků, a prostě jako je problém no, problém pro ty lidi. Prostě najednou dostávají k dimenzi. Znalost toho businessu a podoba toho businessu je právě pro tu firmu strašně důležitá. Takže my se snažíme...máme vzdělávací certifikační program. Certified Logistic Professional, který musí plnit každý zaměstnanec, včetně těch vývojářů. Myslím si také, že je dobré, že když je nějaký vývoj, tak opravdu do toho vývoje zahrnout nějaké vzorky lidí z toho businessu, které dělají přímo v tom vývoje. Což znamená, že tam máte třeba nějakého kurýra nebo nějakého pilota prostě nebo třeba někoho ze skladu, tak opravdu fyzicky

vybrat někoho, kdo to dělá, a opravdu ho do toho týmu zatáhnout. Ideálně kdyby tam mohl být, ale pokud to není možné, tak zvát minimálně na ty retrospektivy, nechat ho vyjadřovat se k tomu, popřípadě to zkoušet sám, protože často i když ten Product Owner má hlubší znalost, nebo stakeholders, se kterými pracujeme, také hlubší znalost...přece to zatažení toho člověka do toho businessu je strašně užitečná věc.

### 13. Jak často se stává, že Product Owner nebo Scrum Master má pracovat s více než jedním týmem? Může to způsobit problémy v projektu?

Já musím říct, že u nás se to stává docela dost. A myslím si, že to není úplně dobře. Já si myslím, že Scrum Master by neměl dělat víc než s jedním týmem. Myslím si, že by měl Scrum Master soustředit na jeden tým. Co se týče Product Owneru, tak tam si myslím Product Owner může pracovat víc než s jedním týmem, třeba dva, maximálně až pět týmů. Jako...tady je jeden Product Owner, schopný uřídit. My jsme chtěli nasadit něco, čemu se říká Scaled Agile Framework, prostě takový Scrum Scrumů vlastně. Že jsme měli třeba 50 vývojářů na nějakém programu a rozdělení na ty jednotlivé scrumové týmy, které vyvíjely určité komponenty a pak vlastně by jako by byly scrumové reprezenty těch Scrumů [smích] a ti formulovali ty požadavky a vlastně to bylo koordinováno takovou tou hierarchií a fungovali celkem dobře. Hodně to bylo myslím schopnější reagovat na ty změny než ta klasická waterfallová metoda. To by si myslím ten projekt dávno havaroval. Pokud bysme neaplikovali tuhle věc. Samozřejmě tam byla taková ta tendence vytvářet větší týmy, ale to takhle nefunguje. Samozřejmě je to těžký. A to je strašně důležité, aby ten Scrum Master nebyl Product Ownerem. A aby byli oba v tom týmu. Protože pokud to tak není, tak to je vždycky zdrojem těch problémů. Já myslím, že pokud už je tým vyzrálý a třeba ten Scrum Master odjede na pár dní na dovolenou, tak to není problém, protože ten tým má, řekněme, ty pravidla hry zažitá a už je sladěný. A když ten tým je formovaný, tak ta role Scrum Mastera je nezastupitelná, protože jakoby staví ten tým a ladí ten proces. Zajišťuje třeba, že když někdo řekne, že ta story je hotová, že je skutečně implementována a testována, a to definition of done, „done, done, done“, jak se někdy říká [smích], tak je tam strašně důležité dodržovat ty pravidla. Protože v okamžiku, kdy lidé uhýbají z těch pravidel, tak pak najednou říkají: „Aaa, Scrum nám nefunguje, protože...“, jo. Často je to prostě kvůli tomu, že se ti lidé ten Scrum prostě nedodržují. To je podle mě na řadě článků, kde lidé kritizovali Scrum, ale já si myslím, že to je často nefér. Protože to, co tam popisují, je to, že té týmy prostě neadoptovali Scrum. Samozřejmě je to extrém a je možné, že to není možné. Já nechci tyhle týmy z toho vinit. Pokud Scrum se podle mého implementuje správně, tak nemá vlastně šanci havarovat [smích]. Protože je to jako navigace v autě, jo. Když máte jako rychlou navigaci, tak ta navigace vám umožní velmi rychle předpovídat ty překážky, které máte před sebou.

Tak můžete jim vyhýbat. I když více méně něco objeví, tak to dokážete objet a tak, jo. [...] Z teorie systému ta regulační smyčka je daleko kratší u toho Scrumu, a tudíž ten systém je daleko adaptovanější. [...] Já si myslím, že typická věc, o které se říká, je ScrumBut. Jako „co děláte? – Scrum, but...“. Nemáme pravidelné iterace? Na vysvětlení: „Scrum, but...“. A jaká je velikost týmu? 15? No, tak to nejste. Nebo...“Scrum? Tak máte Product Ownera? – Ne, a proč?“. A to jsou věci, kdy prostě vidíte, že ty základní principy Scrumu tam nemůžou fungovat. A pak oni řeknou: „No jo, my jsme to aplikovali, ale nešlo to“ [smích]. A prostě to není implementováno.

#### 14. Vyžadujete od PO a SM nějaké certifikáty? Pokud ano, jaké?

Já si myslím, že ty certifikáty skutečně z toho člověka Product Ownera nebo Scrum Mastera neudělají. Ale myslím si, že ty certifikované tréninky, které jsou k dispozici – Certified Scrum Master, Certified Product Owner, nebo i ty vyšší úrovně, pro ty trenéry, jsou velmi kvalitní. Já jsem měl třeba možnost mít velmi dobré učitele. Já jsem prošel všemi možnými Scrum Masterama a Product Ownerama [smích] a tak dále...a musím říct, že to je velmi šikovně, to pomůže, ale to je moc drahá věc. Když máte tisíc vývojářů, tak to je docela moc peněz no. Ale myslím si, že to opravdu může fungovat v těch firmách tak, že pár lidí projde tím školením a pochopí tu podstatu toho Scrumu. A jsou v té firmě prostě nějakým způsobem známé, mají ten vliv. Tak jsou schopní z toho začít v nějakém malém týmu a ukázat té firmě, že to funguje i v té firmě, a ti ostatní to vidí, a udělá se z toho třeba nějaká velká success story. A ti ostatní to vidí, a to chtějí dělat taky. A obzvlášť když to ten management podporuje. My jsme vlastně začínali v NN před 12 lety a „Scrum“ tam bylo zakázané slovo, jo. Tam všechno bylo waterfall. Takový hardcore, pro všechny kroky byly obrovské dokumenty, templaty atd. To bylo neobvykle, že třeba SRS byl několika desetinné stránkový dokument. To nikdo nečetl [smích]. A pak se ten vývoj začal a oni zjistili, že ten zákazník chce něco úplně jiného. A tohle změnit...tak to byl docela problém, jo. Ale dá se to. Myslím si, že to je otázka podpory toho managementu. Myslím si, že tam ten management hraje docela důležitou roli. Ale v okamžiku, kdy tu podporu máte, tak ti lidé mají i radost, že to dělají, protože z toho mají pocit uznání... ale myslím si, že v okamžiku, kdy to začínají používat, tak vidí ty výhody. Pokud skutečně to nasadí správně. Někdy třeba...pokud je to direktivní, tak je to trošku nešťastné, protože ten tým říká: „Tak my to nechceme dělat, ale uděláme“, jo, takže prostě zavedou ty různé ceremonie, ale pořád jsou v tom myšlenkovém waterfallu. To znamená, že třeba dělají v jednom sprintu analýzu [smích], v jednom sprintu – vývoj atd. A dělají stories třeba jako „Analýza: část I“, „Analýza: část II“. A to není story, z toho hlediska, jak je definována. Takže to není úplně šťastná věc. Ale myslím si, že právě ta podpora, podpora nějakých koučů nebo prostě lidí, kteří

jsou dobří trenéři, potom čistě takovými návštěvami v těch týmech upozorní třeba na ty odchylky, tak ten tým se srovná. Myslím si, že to není nezbytně nutné, aby každý tým byl certifikovaný jako Scrum Master, nebo Product Owner. My jsme třeba vytvořili vlastní jakoby minicertifikát a máme takový trénink, kde je to prostě jako už formálně udělané, hezký. Máme jako určitou hru s Legem. Ten tým projde tím vlastně během jednoho dne a základní mechaniky toho Scrumu dokážou naučit. Samozřejmě to nestačí na zažití, ale myslím si, že ten tým toho stačí nakopnout a myslím si, že pak ta supervize nebo navštěvování, kdy ten tým vlastně startuje, někým, kdo už to zná, je podle mého efektivnější, nežli to certifikování. Že se ten tým učí na reálném příkladem.

#### 15. Na kolika agilních projektech může zároveň pracovat jeden člověk?

Já si myslím, že je dobré, aby se člověk dokázal soustředit. Já si myslím, že je nejlepší, když může pracovat na jednom projektu [smích]. Ale to, co jsem zažil v NN, tak je to situace, kdy lidé pracovali na více projektech. Myslím si, že to není dobrá praxe, člověk by se měl tomu vyhýbat. Ale někdy se tomu vyhnout nedá. Někdy jsou třeba role, které jsou tak vzácné, nebo že to by se nevyplatilo, aby tam ta role byla v týmu celou dobu. Protože třeba máte nějakého UI/UX designera nebo nějakého architekta, security experta, a toho vlastně můžete do toho týmu „koadoptovat“ po nějakou dobu [smích]. A pak ten tým zase opustí, jo. To neznamena, že to není nezbytně nutné, že ten tým nemusí být stabilně fixní, ale samozřejmě stabilita toho týmu je hrozně důležitá. Moje zkušenost byla taková, že jsme měli na začátku lidí, kteří pracovali na deseti projektech. A to prostě nefunguje – vůbec. Ale když máte nějakého testera, tak ten je navyklý přecházet mezi těma projektama. Někaké deskriptory, manuální desky... Ale ta doba už je pryč. Dneska s nasazením automatického testování, kde vlastně ta role lidí, co dělají ten test, testový management, je zase posunutá. To jsou spíš programátoři nebo ti lidé, kteří navrhují design těch testů, že jo. Takže vlastně aby jen kódovali, aby dělali mechanické kroky.

#### 16. Popište prosím, jak probíhá komunikace mezi členy týmu.

Já si myslím, že trochu zaleží na formách...na té kultuře v té společnosti. Myslím, že když ta komunikace je víc neformální, tak ten tým funguje daleko líp. Já si myslím, že jsou takové základní principy, třeba „lidé jsou důležitější než věci“ [smích], nebo „komunikace je důležitější než psaní nějakých dokumentů“ atd. Tyhle ty věci jsou hluboké pravdy. A myslím si, že pro tuto spolupráci je to strašně důležitý. Já si myslím, že když se vzájemně respektují a když je ten Product Owner skutečně náročný, ale zase ten Scrum Master, ten tým dokáže to ochránit a zajistit mu normální podmínky, tak potom ta komunikace v rámci tohoto týmu

funguje samozřejmě. Ti lidé vytvoří jakousi váhu, rodinku...[smích], nějakou sociální jednotku, která prostě jednou jako by začne mít společné eso. Nedávno četl nějaký článek o teorii integrované informace, která vlastně popisuje podstatu vědomí, a je zajímavé, že v případě té teorie je víc prvků systémů, které dokážou integrovat stejnou informaci, tak vlastně vytváří zárodek toho vědomí. Máte nějaký zážitek a je to prostě vědomí. A když ten tým má stejný zážitek, tak vzniká skupinové vědomí, a tím vysvětlují chování davu a prostě lidé na koncertech a podobně. Ale já si myslím, že to je i ta, co se děje v tomto Scrumu. Nejenom když se ti lidé naladí na jednu vlnu. Vlastně komunikují velmi, velmi efektivně. Stačí něco zmínit, a ten druhý člověk jako tuší, co ten druhý člověk chtěl říct. [...] My máme hodně lidí po celém světě, a ne vždycky je možné ten tým mít jenom na jedné lokaci. Takže musíme mít nějaké...ale i tak ta spolupráce s businesssem...často třeba business v Americe, nebo v Anglii a tak ve světě...musíme mít nějakou základnu, kde prostě vidíme ty náklady, jsme schopní plánovat, a tak demonstrovat... Třeba jsme dělali takovou tu alokaci různých nástrojů. Ve firmě standardně používáme regional qualified management, DOORS...to jsou nástroje, které umožňují definici požadavků, plánování, trackování, testování, test management. Já musím říct, že té nástroje jsou nesmírně flexibilní, a trvalo nám dost dlouho, než jsme se naučili, jak je používat. Myslím si, že spousta menších nebo středních firem používá třeba Jiru. Ta, si myslím, je velmi populární, ale třeba před těmi deseti lety ta Jira nebyla vůbec zralá na nasazení v enterprise prostředí. Za těch 12 let ta Jira hodně dospěla a my máme teď kon často i kombinaci, třeba...jak jsem říkal o tom supply chainu, tak třeba ten projekt se skládá z řady několika týmů, máme průzkumní, vývojový týmy, a ti třeba používají tu Jiru, a my třeba...ten quality development používá RTC management, a tam je nějaká synchronizace, nějaké synchronizační, integrační projekty... Nejen synchronizovat ty stories nebo ty defekty, včetně attachmentu atd., ale i těch stavů, takže když ten vývojář pracuje...fixne nějaký defekt, udělá „fixed“, tak v tom RTC se to promítne. Takže...je to fakt dobrý. Jsou ještě i další nástroje, které jsme zkoušeli využívat, ale myslím, že ta Jira má asi největší pokrytí toho trhu, a i u nás jako...z většího jako ty týmy postupně i přecházeli na tu Jiru. Ale to prosím Vás necitujte [smích]. Náš standart je Rational Team Concert. A ten má prostě takovou výhodu, že pokud je plánování a trackování agilních vývojových projektů, tak to je product placement i tradičních projektů, takže to můžete mít i jako waterfall projekt, a i v té Jire taky. Protože je docela dobrý. A jsou v tom nějaké škály a tak. Takže si myslím, že je to dobrý no.

17. Používáte nějaký software k plánování sprintů, meetingů, pro backlogy atd.? Pokud ano, uveďte název.

Právě pro nás ta RTC je takový nástroj. Rational Team Concert. Nástroj, který používáme pro plánování a trackování. Prostě tam je koncept backlogů interně zabudován, takže je to takové přirozené. Ale chvílku nám to ještě trvalo, než jsme to doladili a mohli nasazovat v těch týmech. Ten tým...člověk prostě musí rozhodnout mezi tím jednoduchým nástrojem, který pak není tak flexibilní a neumožní přizpůsobit se speciálním požadavkům v těch týmech, ale tím, že je jednoduchý, tak se snadno nasazuje, anebo prostě mít nějaký flexibilnější a složitější nástroj, který se potom uživateli přizpůsobí. Musí se tomu hodně věnovat, aby se ten nástroj fungoval správně. To je těžký, no. Ale ono se také vyvíjí. Ta Jira před pěti-šesti lety byla velmi jednoduchá. Dneska se stává Jira komplexnější a komplexnější [smích]. Takže je to otázka, jak se to vyvine. Je tam otázka, kdy Atlassian jako taková ta firma směřuje. Ten směr je takový dost do businessu, do business managementu. A je to otázka, nakolik oni chtějí to do budoucna podporovat, takový to trend ve vývoje. Ale to jsou otázky nějakých spekulací. Rozhodně Jira je nejlíp integrovaný nástroj mezi dalšími produkty. Třeba GitHub a další. Zde jde o to, že pro plánování a trackování RTC je věc, kterou používáme standardně. Kdybych začínal znova a neměl žádnou historii, tak asi bych asi zvažoval tu Jiru. [...] Každopádně to je jedna ze silných stránek Jiry, že má velký ekosystém těch vendorů, které dodávají různé integraci s Jirou. A tím, si myslím, že jako by byla cenově dostupnější než ty produkty od IBM, tak vlastně daleko rychleji se rozšířila na těch menších a středních firmách. A pak ten produkt potom dozrál. Ten Rational Team Concert je od počátku budován jako platforma pro koordinaci těch týmů, a je obecnější. A je pobudován na nějaké platformě, které se říká Jazz. To je vlastně jakési base, na který se můžeš nějaký nástroj integrovat. Integrace mezi Rational Team Concertem a DOORS, což je requirement nástroj, je velmi snadná. Nebo integrace s test management. Jira jako sama o sobě nemá test management, není to test management nástroj. Ale do Jiry existují nějaké integrace, které jsou process management, ale...nevím, jak to vypadá teďkon, ale historicky to nebylo tak těsně provázáno, nebo ta integrace nebyla tak přirozená, jak to udělal IBM v rámci svých produktů. Znáte ty produkty...ty rozhraní jsou obecnější no. Ale v zásadě je mi jedno, který nástroj. Já si myslím, že nástroj nehraje tak velkou roli. Já znám týmy, které používají zed' a stickery a fungovali efektivně. Pokud je ten tým v jedné místnosti, tak si myslím, že je to v pohodě. Ale v okamžiku, kdy máte nějaký distribuovaný tým, chcete mít nějakou metodiku, kde chcete třeba i jako sledovat nějaké statistiky, nějakým způsobem reportovat a tak, tak v tom okamžiku nějaký nástroj potřebujete. Znam třeba lidí, kteří používají Excel nebo SharePoint. V SharePointu existuje plugin pro Scrum, a je to velmi primitivní. Ale třeba pro jednoduché scrumové projekty to může i fungovat.

18. Jakou délku sprintu obvykle máte?

Já si myslím, že délka sprintu nikdy nesmí být větší než měsíc. Ideálně do dvou týdnů, možná i tři. Myslím, že tři týdnů je takový optimum. Třeba teďkon mám nějaký projekt, který má dvoutýdenní sprinty. A myslím si, že pro tak šestičlenný tým fungují velmi dobře. Ale tři, když máte nějaký další...no, zaleží, co od toho projektu chcete. Když ten projekt je krátkodobější, tak potřebujete možná i kratší ty sprinty. A když ten projekt je dlouhodobější, tak možná je to někdy možné i natahovat. Třeba jako to, co jsme teďkon rozhodli nasazovat... My vlastně máme dvoutýdenní sprint, ale vlastně každý sudý sprint pozveme víc lidí, jako těch stakeholderů, takže je to jakoby větší sprint review [smích]. Je to vlastně i tak, že to nahrazuje něco jako project board review. Máte prostě možnost i v rámci třebaměsíčních reportingů, které v téhle firmě máme jako prostější stav, který je po tom měsíce. Ale máme ještě jako mezisprint, který vlastně umožňuje tomu týmu reagovat rychlejc. Že ten měsíc už je skoro moc.

19. Pokud v průběhu sprintu nějaký úkol není splněn, co se s úkolem dále stane? Jak často se Vám to stává?

To se občas stane. A typické řešení je, že ta story vstupuje do dalšího plánování. Máme vlastně na konci sprintu nějaký sprint review a tam se to konstatuje v podstatě, že „tuhle tu story jsme nedali“, „udělali jsme to jen částečně“. Typické je se ta story zařadí do plánování, do dodávání dalšího sprintu. Ale neznamená to nezbytně nutně, že v tom dalším sprintu bude implementována. Tam vstupuje další stories a je to otázka priority. To znamená, že ta story, co se nestihla a ukáže se jako méně důležitá než třeba jiné stories, které mezi tím se zformulovaly, tak může být prostě odložená. Nebo třeba na několik sprintů, nebo se i dokonce může dostat z toho releasu pryč. Protože se prostě ukáže, že není validní. To se může stát. Vždycky u těchto nástrojů je taková otázka, jestli ten item nebo story, jako nějaký objekt v tom nástroji, jestli ji duplikovat nebo nechat tu story v původním sprintu a udělat druhou story, která bude jako kopii v tom dalším sprintu a bude jako žít dál. Že jako roztrhnete tu story a to, co jsme udělali, tak to je skupina jedna, a co ještě zbývá udělat, tak to je skupina dvě. A to je jako další story, jo. Anebo prostě vezmete tu story a řeknete: „Ta story není hotová a dokud není hotová, tak ji posunu“. A dokud není hotová, tak není hotová. Já si myslím, že ten druhý přístup je jednodušší a funguje velmi dobře [smích]. Ale nemělo by to být, takže ten tým jen hladce to přejde. A měli by se zeptat, proč to třeba nastalo. Během retrospektivy se zeptat, proč se to vlastně stalo, že jsme to nedodali, jo. Ale to neznamená, že by ten tým měl opravdu trpět nějakou strašnou vinou. Důležitější je, aby se z toho poučil a šel dál.

20. Jaká je střední délka plánovacích meetingů?

[Odpověď je zahrnutá v předchozích otázkách]

21. Daily stand-up by měl ideálně být každé ráno ve stejnou dobu, ve stejném místě, ve stoje a maximálně 15 minut. Jak to máte Vy?

Já si myslím, že je to trochu challenge v našem prostředí, protože já mám speciálně třeba spoustu různých meetingů a ráno hodně komunikuju třeba s tou Asií. Takže nejsme schopní to mít ve stejnou dobu, jo. Ale je důležité, aby to bylo každý den. A jsou situace, kdy třeba ten člověk onemocní nebo prostě nemůže to zúčastnit, a s tím se nedá nic dělat. Ale je strašně důležitý, aby ti lidé byli náchylní, aby daily stand-up probíhal tím způsobem, kdy řekne, co udělal za posledního setkání, jaké má priority, nerozpovídal se moc. Protože v případě, že by tam byly nějaké imperimenty, tak to řešit mimo ten daily stand-up. Nebo už třeba před těma meetingama Scrum Master řekne: „Dobře, tak jste se nedohodli“, nebo „Ty potřebuješ pomoc, po tom stand-upu se sejdeme a ty a ty – my budeme ten problém řešit“. Jinak ten zbytek toho týmu, když ti dva řeší ten problém, tak to natahuje délku toho sprintu. To se zase typicky děje na začátku, když ten tým vlastně adoptuje ten Scrum, protože oni si myslí, že to je takové normální denní kafičko nebo prostě povídání, co kdo dělá a tak, jo. Tam je ta role toho Scrum Masteru neskutečně důležitá, aby ty lidi jakoby upozorňoval a umírňoval, protože myslím, že opravdu stačí 5–10 minut pro pětičlenný tým. Na ten stand-up je dobře, když třeba devítičlenný tým má 15 minut, tak rozhodně víc než 20 minut by to nemělo trvat. Ale myslím si, že těch 15 minut je jako maximum. A myslím si, že i 10 minut je takový optimum, když máte osmičlenný tým třeba.

### **Výsledky implementaci**

22 Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily?

Rozhodně je to ve prospěch práce v týmu. Jak jsem řekl, vlastně ta interakce je daleko intenzivnější v tom týmu. Také si myslím, že ten tým daleko víc spolupracuje nebo je daleko víc propojený. Ze začátku to je možná problém. Někteří programátoři jsou trochu introverti a je trochu problém je zapojit třeba. Myslím si, že je to zvláštní, ale většinou to prostě i nějak funguje i u těch lidí. [...] Já si myslím, že waterfall byl vymyšlen Scrumem. Prostě historický tomu nikdo „waterfall“ neříkal. Prostě to byly nějaké metodiky softwarového vývoje, ale teprve když se vytvořil Scrum, tak vůči tomu se začali ukazovat – to je waterfall. Podíváte do těch původních knížek, tak to pojmenování waterfallu předvídalo určitou iterativnost toho dalšího vývoje. Pokud chápeme waterfall jako skutečně ten kaskádově definovaný vývoj, tak nevidím v zásadě žádnou výhodu použití waterfallu proti Scrumu. Snad asi kromě těch naprosto jasných požadavků a naprosto známých technologií, kdy ten tým je jako hodinový stroj, který vlastně



nepotřebuje nějaké ceremoniály. Ale já si myslím prostě, že ty sessions, jak jsou definované v tom Scrumu, hrají velkou roli, a lidé třeba, nebo i ten management říká: „Vy se pořád setkáváte a pořád něco kecáte“ [smích]. Pravda je ta, že i když to vypadá hloupě, tak to je fakt strašně důležitý. A myslím si, že to je důležitější než tisíc nějakých dokumentů nebo nějakých strašně složitých procesů nebo expertů a tak.

### 23. Mělo zavedení agilu dopad na rozsah dokumentace k projektům?

Bezpodmínečně má na tu dokumentaci, ale myslím si, že jedno z velkých nedorozumění je, že ty týmy často říkají, že Scrum nevyžaduje žádnou dokumentaci, že stačí psát jenom story a je to. Ale to není správně jako, protože nezbytnou součástí toho produktu musí být nezbytná dokumentace, která je potřebná pro support dlouhodobě. A to je prostě role Product Ownera, aby to zajišťoval. To znamená, že ty dokumenty jsou součástí toho produktu. Včetně jako designu nebo architektury. Security, design – to je třeba důležité třeba pro certifikace, kterou ten produkt musí projít, pokud má splňovat nějaké požadavky, nasazení v nějakém enterprise prostředí, jo. Nebo, co se týče nějakých produktů, které mají uživatelské rozhraní: nějaké mobilní aplikace nebo webové aplikace, tak musíte mít aspoň nějakou uživatelskou příručku nebo aspoň nějaké základní informace o tom, jak to stojí efektivně používat. Nebo nějaké tréninkové materiály, já nevím. Tyhle tedy dokumentaci jsou prostě nezastupitelné. Třeba nějaký instalační guide – to jsou prostě dokumenty, které jsou validní a nezávislé na tom, jakou metodologii máte [smích]. A má smysl vytvářet ty stories, vytvořené třeba několika řádkami a to stačí. Na druhé straně, a to je možné, že ty týmy nedokážou tak rychle dodávat, když třeba takovou dokumentaci nevytváří, když vidí, co ty stories přesně znamenají. Ale má to potom jinou nevýhodu. Když někdo nový do toho týmu přijde, nebo se něco stane...prostě někdo z toho týmu odejde, tak pak tahle znalost zmizí a může to být třeba problém, jak tu mezeru nebo prostě neznalost vyplnit. Tak když ten člověk pak čte nějaké stories a nezná ten kontext, tak třeba vůbec netuší, o čem to je. A je to důležité i z hlediska toho supportu. To znamená...představte si, že ten produkt se skládá z několika produkcí a najednou přijde nějaká chyba a lidé se snaží zjistit, kde ta chyba je a jak ji řešit. A když není ta dokumentace, tak není jasné, které vlastně řádky toho kódu co znamenají [smích]. Tak to je těžký, no. Ale já si myslím, že důležitá je dokumentace přímo v tom kódu. Myslím, že to je jako ta nejcennější věc – kvalitní kódování a prodlužování nějakých jakoby dobrých best practicies. Dneska je trend využití nějakých modulárních vývojů, použít nějaké micro services. To jsou prostě prvky, které pomůžou flexibilně definovat ten systém a pak ty specifikace jsou jako víc discribované, autonomizované. A máte v podstatě takovou sémantickou síť té funkčnosti toho produktu, spíš než nějaký velký dokument, který prostě obsahuje tu věc. Tak vlastně když budete mít nějaký

dokument třeba System Requirements Specification, kde jakoby jsou požadavky pro každou story a každý odstavec, tak dodržovat je je prostě noční můra, jo. Ale právě použití requirements jako nástrojů jako DOORS. Ty právě umožňují ty requirements provázat do nějakého dokumentu nebo virtuálního dokumentu, kde je to vlastně normálně čitelné tím analytikem nebo tím zákazníkem, a přitom je možné si odkazovat a pracovat s těmi požadavky anatomicky a prioritizovat je z různého pohledu atd. Ten requirements management je tam hrozně důležitý. A to je právě ta otázka: kde, co a jak definovat. Jsou jakoby dva základních přístupů k tomu requirements managementu. Ten tradiční, kdy máte od nejvyšších úrovní abstrakce...business requirements, na system requirements, na design atd. A to, jak se vytváří ta dokumentace. Nejdřív ta obecnější, pak přesnější a přesnější. Anebo máte takový agilnější přístup, kdy vlastně máte nějaké stories, kde definujete jednotky té funkčnosti, fičury, které přináší nějakou hodnotu a můžete to elaborovat. To znamená, že když jim nestačí nějaký jednoduchý textový popis, tak můžete jakoby napsat přesnější popis, nebo prostě definice nějakých business pravidel a to vám už nestačí ten jednoduchý textík. Tady máte prostě v tom description té story, a tam, myslím si, že jsou třeba možnosti použití nějaké Wiki, nebo nějaký další nástroj než dokumenty jako takový. Lidé třeba používají Confluence v kombinaci právě s Jirou, právě tu sémantickou síť těch požadavků. Právě ta Wiki se dokáže zachytit velmi dobře. A Wiki obecně má trochu problémy s review a tak. [...]

#### 24. Bylo podle Vašeho názoru zavedení agilu přínosným pro Váš podnik?

Rozhodně jo. Myslím si, že bez toho ta firma nemá šanci uspět na tom trhu, který se neustále mění. Ty technologie se vyvíjí tak rychle, že prostě by nasazování přístupu waterfall vedlo k totálním haváriím. Konkurence to dělá a my nemůžeme prostě lít do toho trhu, prostě zaostat. To si nemůžeme dovolit. A myslím si, že to jde i dál, že ten Scrum a agilní metodiky vývoje není základ. Já se domnívám, že mnohem důležitější, to, na čem je nutné soustředit si, je ten DevOps, propojení developmentu a operations. Vlastně rozšíření těch agilních přístupů i na ten support toho těch aplikací. Takže ti vývojáři jsou víc vtažené do toho, co se děje v té produkci, a vidí ty chyby v té produkci velmi rychle a jsou schopní daleko víc na to reagovat. A zase ti lidé, co zajišťují ten support té aplikace, můžou obohatit vývoj toho týmu s tím, že prostě vidí, co se děje v té produkci, a můžou přicházet s nějakými nápady. Co by bylo dobrý, co zlepšit, jak zlepšit tu podporovatelnost té aplikaci, protože řádu se dá automatizovat, ale třeba programátoři si to nedokážou ani představit. Ten tým je víc propojený, a tím daleko efektivnější. Ale zase, to je prostě taková bariéra, která v těch firmách je tradičně vybudována, protože ti vývojáři chtějí změnu, a lidé v těch operations změnu nechtějí [smích] a tohle propojit je velká challenge. To je to, čím teď vlastně procházíme jako firma. [...]

## Hodnocení a doporučení

25. Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.

Já si myslím, že ten pozitivní dopad je v tom, že ti zákazníci dostávají to, co opravdu chtějí. Jako to zní hloupě [smích], ale často se s tím tradičním waterfallovým přístupem se stává, že se tomu zákazníkovi na začátku moc neporozumělo, takže vlastně těch requirementů nadefinoval spoustu, ale obecně. A pak se postupně zjišťovalo, že zákazník to nechce, ale vlastně i v tom designu, i v té implementaci bylo zbytečně některá řada requirementů, které nepotřebuje úplně, jo. A ten Scrum, si myslím, právě tam pomohl v tom, že zajišťuje tu největší přidanou hodnotu a je skutečně dodána. A ty méně důležitější nebo méně riskantní věci jsou vždycky později v tom backlogu, později i v tom vývoji. Co znamená, že je to způsob, jak optimalizovat nebo maximalizovat tu přidanou hodnotu toho daného produktu. Myslím si, že je to nejdůležitější. Myslím si, že to pomáhá i v zvýšení rychlosti toho týmu, ale nesmí se to přehánět. Myslím si, že to je prostě „side effect“ a ta produktivita a celkově ta atmosféra v té společnosti se daleko zlepší, ale není to jako nastavovat nějaká KPI...že to není jakoby cílem. Cílem je skutečně dodat ten produkt, který ten zákazník chce. To si myslím, že Scrum nabízí takovou možnost. Já si myslím, že je to vždycky o lidech. Možná, že jako dobrý příklad těch benefitů... jsem zažil projekt třeba v Malajsii a ten projekt byl vyvíjen waterfallem. A tam ten tým byl asi půlroku a specifikoval nějaké požadavky. 200 strankový requirement specifikace. Tak ten poslali prostě zákazníkovi, a on říká: „Tomu vůbec nerozumím, to jsou nějaký diagramy. Já to vůbec neumím“ a tak [smích]. A pak to teda vlastně vedlo k tomu, že ani nebyly schopní se dohodnout na tom, co vlastně ten tým měl přesně udělat, jo. A nakonec to vedlo skoro k velké krizi. Nemůžeme vyvíjet jakoby obecný produkt, ale když to začali vyvíjet, tak ten zákazník řekl, že tak to prostě nejde. A pak jsem řekl: „Tak dobře, tak zkusíme Scrum“. A když jsme to použili, tak najednou všechno začalo fungovat. Ten zákazník ze začátku jako vůbec nechtěl komunikovat. „Já přijdu, až to budete mít hotový“, jo. A my jsme řekli: „Dobře, tak zkuste přijít na ten náš sprint review, čistě jen pro zajímavost“. A to byla nějaká mobilní aplikace, abychom zjistil mobilní číslo toho člověka z toho businessu, a když jsme přímo v tom sprint review ukázali tomu člověku přímo prototyp, že jsme simulovali nějaký user case, tak on dostal tu smsku přímo v té mobilní aplikaci. A v tom okamžiku vlastně zjistil: „Ach jo, tady je něco, co bych chtěl trochu změnit“. A od té doby ho to zatáhlo, a on byl na všech těch sprintech a teď je velkým propagátorem Scrumu ze strany businessu, jo. A všechny další projekty chce mít ve Scrumu, jo. To je takový zářivý příklad.

26 Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?

Já si myslím, že nevýhody...že se to skutečně vyžaduje určitou disciplínu, jo. A ne každá skupina, ne každá společnost je schopná toho dosáhnout. Pokud není schopná dosáhnout, tak si myslím, že může trpět. Myslím, že by v té situaci to nemuselo být tak výhodné. Nebo pokud, já nevím, dlouhodobě dělají na nějakých technologiích, mají zcela dlouhou dobu, dělají to prostě tradičně a vědí, že ta práce trvá dva man/dayi, jedna fáze je čtyři man/dayů. Velikánský kaskádový model, dobře vyladěný, tak si myslím, že tam mohou být daleko efektivnější, jo. Ale myslím si, že speciálně v těch nových technologiích... Nevidím žádný důvod, proč by neměl Scrum být používán.

27. Co byste doporučoval/a lidem, kteří chtějí dané metodiky implementovat?

Myslím, že to je otevřená mysl a skutečně nechat se buď jako vyškolit, nebo ideálně zúčastnit se nějakého existujícího projektu. Protože to, že člověk to zažije s někým a pochopí ty mechaniky, jádro toho Scrumu, to je strašně důležitý. A myslím si, že je strašně důležitý nepodceňovat ty role a ty základní pravidla toho Scrumu. To je jako fotbal prostě. Když člověk nedodrží pravidla, tak je to chaos, jo [smích]. Prostě to se týká těch definicí těch stories, jo. Musíte mít pro každou story definition of done. Musíte mít ty jednotlivé meetingy, jednotlivé aktivity, aby ty stories byly přepravené pro další sprint. Zajistit, protože jinak ti lidé začnou běhat, stornovat na sprint planning meetingu, a to už je potom pozdě, jo. A ty týmy, ty meetingy jsou potom neefektivní. To se dá všechno připravovat. Když se to takhle nastaví, tak myslím si, že to jde. A myslím si, že jako to nevzdávat. Po nějakém proudném nadšení přichází takový propad, když zjistí, že to bolí, jo. A v ten moment prostě to nevzdat [smích]. A myslím si, že když ten člověk to nevzdá a ten tým jako by tou těžkou zkušeností projde, tak najednou ta produktivita a ta efektivita toho vývoje poletí nahoru.

28. Chybělo Vám něco v našem dotazníku? Chtěl/a byste jej nějak doplnit?

Myslím si, že jste nezmiňovala určitý trénink, který se objevuje v tom agilním vývoji. Právě jsou ve velkých společnostech nasazení Scrumu trošku problém, jo. Když máte větší týmy nebo větší produkty, tak máte třeba požadavky na daleko rychlejší vývoj, a tam je komponent, na kterém se třeba podílelo hodně lidí, a tam určitě musíte mít víc těch scrumových týmů, a jak tohleto koordinovat? V poslední době se vyvinulo, vytvořilo se hodně metodik. Myslím, že nejpopulárnější je Scaled Agile Framework, SAFe se to jmenuje, a tohle je podle mého věc, která je moc populární, ale jsou i další, které si myslím jsou daleko slibnější. Jako např.

Discipline Agile Delivery. To jsou prostě různé evoluce toho Scrumu pro enterprise prostředí, takže té termíny jsou trošku přejmenované. Ale ty jádra, ty principy jsou prostě v tom Scrumu. A...jenom třeba tu koordinaci mezi těmi týmy, to Scrum normálně sám sobě neřeší. A to je taky vazba mezi projektovým managementem a tím scrumovým týmem. Jedna věc, co jsme zažili...měli jsme tým...Project Management Office...a ti projektoví manažeři...když jsme najednou přišli s tím Scrumem, tak měli strach. Při jejich práci...protože vlastně ve Scrumu není žádná role projektového manažera [smích]. Ale pak se ukázalo, že to není problém. Protože z hlediska PMP nebo Project Management Practicies...PMBok...nebo prostě dalších metodik, těch metodik je víc. Třeba Prince 2 je definován na vyšší úrovni, jo. To znamená, že Prince 2 nevyklučuje Scrum. Třeba měli jsme na začátku takovou diskusi, jestli Prince 2 není v rozporu se Scrumem. A když jsme to analyzovali, tak jsme vlastně zjistili, že přesto se to dá udělat. Že ten Scrum je o úroveň níž. Ty prvky, které se týkaly managementu, těch nákladů na tom projektu, project board atd., to pořád dává smysl, jo. Ale fakt je ten, že ten projektový manager nemá tolik práci s tím mikromanagementem. Někdy třeba bývá situace, když ten projektový manager je schopný pochopit ten Scrum a stává se třeba Scrum Masterem. Řada dobrých projekt managerů se u nás stala, definuju, Scrum Mastery. A myslím si, že to je docela dobrý model. Ale...skutečně to jsou tyhle role. To znamená, že třeba ten člověk může mít dvě role, anebo může jeden projektový manager mít několik projektů...které jsou scrumové...a v tom by neměl být problém. Protože on se vlastně stará, aby se ten projekt držel v rámci těch project boundaries, rámcového a časového scope tohoto projektu. A když to hází nějaké riziko, tak třeba ten rizikový management Scrum neřeší moc, jo. A pak vlastně ty nějaké eskalace, nebo ty řešení třeba v rámci toho project boardu – to prostě Scrum neřeší, ale je to důležitá část, pokud máte nějakou velkou společnost a chcete to prostě více propojit. To je asi tak všechno. Jo, ta automatizace. Myslím si, že ta automatizace je čím dál tím větší. Má to větší důležitost. A myslím si, že právě pro aplikování těch agilních metodik je strašně důležité. Že ten tým co může, zautomatizuje a funguje daleko efektivněji a ti lidé se můžou soustředit na tu kreativní část té práce. My třeba máme ty nástroje...jako služby, v rámci společnosti, co znamená, že ten Rational Team Concert, nebo i ty vývojové nástroje: audit nebo Jenkins atd., ty poskytují služby. A ty vývojové týmy je můžou využít okamžitě. A to je prostě výhodná věc, že můžou začít velmi rychle. A když to vytvoří...nechci říct česky...pain plan. Provázání těch jednotlivých kroků, které jsou provázány, od změny toho kódu až po ten deployment a testování v těch různých prostředích a...může to projít i několikama prostředí. Třebamáme týmy, které dneska vlastně kontinuálně mění kód, a ten kód sám automaticky dostava do development prostředí, prochází nějakými automatickými testy. [...] Stakeholdeři

třeba na to dívají, a to běží je prostě automatický. A pak mají i automatické deploymenty té produkce, což třeba na jednu strašně zjednodušuje a snižuje ten stres toho týmu. Když je ta věc automatizována, tak je to skutečně “push button“ [smích]. Že ten člověk zmačkne to tlačítko a dostane se tam, do toho prostředí, a ten feedback, ta zpětná vazba, smyčka toho Scrumu, a i toho feedbacku, který má se s tím zákazníkem, zkracuje strašně rychle, a zvyšuje to i tu kvalitu prostě. Protože ty problémy deteknou častěji. Shift+left jako. Čím dřív se to objevíte, tím je to lepší. Já bych mohl mluvit hodiny, ale...[smích]. [...]

## **Příloha 4. Přepis rozhovoru s respondentem č. 3**

V tuhle chvíli vlastně ve firmách nedělám žádné. Dělán na sebe, projekty na zakázku, které teda samozřejmě taky řeším nějakým agilním způsobem, ale ne Scrumem. Vlastně do léta jsem dělal v tady jednom startupu, kde jsme měli samozřejmě zavedlý Scrum a do jara tři roky tak jsme dělali v XXX, tam jsme na Scrumu jeli vlastně dva roky, rok a půl, abych...to teďka nevím přesně. Co teďka já vlastně rozjíždím...nové herní studio, trochu jinak jakoby směřované, než jsme to měli prvé. A tam to [agile] samozřejmě bude taky, protože já jsem toho velký zastánce. Vím ze zkušeností, že prostě tak to má být. Já jsem před asi třemi, čtyřmi roky pracoval tady v jedné plzeňské firmě, kde jsme jeli na Scrumu tři roky, tak jako ve velkém týmu, takže vím, jak to funguje, vím, jak se s tím má pracovat. Samozřejmě výhody, nevýhody...tak bych řekl, že to mám v malíku teda.

### **Obecné (úvod)**

#### **1. Připadají Vám agilní metody přínosné, nebo ne?**

Samozřejmě že jsou přínosné. Jako právě největší problém tohohle je přechod firmy, která takhle nefunguje...nebo celkově – i prostě nějaké skupiny lidí...budeme tomu říkat „firma“...vlastně, nejtěžší je ten přechod. Ať už prostě ty lidé dělají to z nějakého donucení, z nějakého přesvědčení, spíš je to těžký. A to, co slyší vlastně manažeři a všichni takoví ty kolem toho, nejvíc jakoby co slyší nerádi, že to zpomalí tu práci. V prvních několika prostě iteracích...co to zpomalí, ale ty výsledky jsou potom úplně někde jinde a takové ti lidé nikdy neviděli.

#### **2. Proč je dnes důležité se zabývat agilními metodami?**

Měly by se tím firmy zabývat. Nejlepší na tom je, že to jde prostě nahrnout na všechno: ať už na vývoj softwaru, nebo, já nevím, na výrobu v automobilce, ze které to mimochodem vychází. Tak jde to dát všude. Tam je jenom důležitý nastavit si k tomu správný proces a hlavně mít ty kolegy a celkově takový kolektiv, který se s tím dokáže vyrovnat a chce hlavně pracovat.

#### **3. Jaké výhody a nevýhody mají podle Vašeho názoru mají agilní metody v porovnání s waterfallem?**

Takhle, oproti vodopádu nebo waterfallu, jak to máte, což jako v dnešní době je neskutečně zastaralá věc...nemělo by to prostě být. Napříč prostě odvětvím to by být nemělo. Každopádně nepřesvědčíte...já nevím, tady Komerční banka, která je šedesátiletá, aby přešli na něco takového, to snad ne, to se prostě nestane. Každopádně obrovská výhoda je abso...teď to možná

bude znít trochu...jako...nechci říct „fašistický“, ale úplně šíleně...obrovská výhoda je skoro stoprocentní kontrola nad tím, co vyrábíte. Já osobně nesnáším nějaký mikromanagement nebo něco takového, ale je prostě vidět, jak se něco vyvíjí...já nemám rád to „vyrábí“...prostě jak se něco vyvíjí od počátečního nápadu až po finální produkt. Celkově v tom agilním vývoji ten krok, každý ten krok můžete sledovat. Samozřejmě Scrum jako takový je na to přímo dělaný, ale když se používá klasický Kanban, tak tam taky to pořád jde. Máte tam vlastně kontrolu nad tím, co se dělá. Jenom nemáte v tom tu časovou hodnotu. To je obrovská výhoda. Další výhoda je to, že zrychlí způsob té práce a ve výsledku je to volně rychlejší než cokoliv jiného. Pokud tam jsou správné akceptační kritéria a takovydle věci, tak je to...řeknu prostě o 60–80 procent rychlejší než cokoliv jiného. Teď mluvím o Scrumu, jo. Celkově agilní přístup jako takový...to nevím o kolik zrychlí a zrychlí to, samozřejmě. Nevýhoda toho je, jak už jsem říkal na začátku, že to má pomalý rozjezd. To znamená, že naučit se to, vychytat správně všechno a dát to dohromady, tak chvíli trvá. Co se s tím dá, je to, že jakoby...když to řeknu blbě...tak jakoby ten management a tohleto...tak v těch firmách jakoby s tím úplně nesouhlasí, protože ve dnešní době se všechno chvátá a chvátá, a musíme to rychle mít hotový, aby se nám udělaly peníze, což, samozřejmě, dává smysl, protože bez peněz ta firma nemůže existovat a nemůže to dělat. Tady je důležitý najít nějaký kompromis – při tom přechodu.

## Scrum

### 4. Máte zkušenosti s metodou Scrum?

V tom mám hodně, to už jsme si řekli. Každopádně, ono dneska...já jsem spoustu klientů řešil, že se jim nutí spousta lidí, kterým právě nabízí možnosti Scrumu. Právě...no to teda není úplně odpověď na tu otázku...no jako je, spíš prostě se Scrumem ve dnešní době jakoby z objektivního hlediska jsou takové, že spousta lidí na to chce přejít, ale nerozumí tomu. Využívá to spousta lidí, kterým to nutí, a ty výsledky jsou pak strašné. To jsou...to je asi moje nejhorší zkušenost...že se Scrumem jel jeden takový typ...prostě dohazoval jednomu člověku, který má jednoho člověka na vývoj, což...tak se to nedá dělat. Každopádně, já mám zkušeností, mám...jak jsem říkal, dělal jsem prostě ve třech firmách. Scrum jako takový a Kanban pak ve dvou. Teďka samozřejmě ta nová firma bude vyjíždět také na Scrumu.

### 5. Uveďte prosím základní odlišení Scrumu od jiných agilních metodik.

Tak základní odlišení jako takové jsou iterace. V podstatě oproti Kanbanu je to tak, že vlastně Kanban jako takový nepracuje s časem. Jako...má nějaké deadliny a podle to, ale to je cesta do pekel. A Scrum pracuje s nějakou časovou jednotkou, což je ten sprint, nějaké trvání té



práce. To je takové největší odlišení Scrumu od ostatních metod. Je to pak ve výsledku dost podobný. [...] Ono to dá do Scrumboardu. Praktické Scrum z toho Kanbanu vychází. Jediný rozdíl, že tam přidáváte ty sprinty, zjednodušeně řečeno. Což pak ještě je na Scrumu super, to samozřejmě počítání velocity a takovýchto věcí, to je samozřejmě kapitola samá pro sebe...tak to je další velké odlišení. Ale na druhou stranu jsem zažil spoustu firem, co bodování už přestalo používat, což mě docela fascinuje, ale je to otázka.

#### 6. Čím je možné vysvětlit takovou popularitu Scrumu?

Já už jsem to nastínil. Je populární proto, že se z toho stalo novodobé klišé. Spousta firem křičí...ted' mluvím jakoby z toho našeho oboru, z toho technologického oboru...spousta firem křičí, jak na to chce přejít, ale neví, kde začít, neví, jak to mají správně dělat a jak to funguje. Proto je tak populární. Dneska vlastně kdo prostě není „agile“, kdo nemá Amazon Web Services, tak je mrtvý. Samozřejmě s nadsázkou řečeno. Takže to je, proč je tak populární. Já si myslím, že kdyby to všichni dodržovali tak, jak mají, tak... není populární a neděje se to tam, kde se děje, kde se prostě různí klienti vyžadují, aniž by věděli, co to pro ně znamená. Což asi není odpověď, kterou jste chtěla slyšet [smích]. Já tohleto o tom opravdu myslím a bojuju s tím dost často, no. [...] Je to smutný, ale je to tak.

#### 7. Napadá Vás příklad jiné agilní metodiky, která je vhodná k implementaci?

Jak jsem říkal, Kanban. Ale...ono nehodí se pro všechno. Tam se to hodí pro projekty...když to řeknu úplně jednoduše...Kanban je takový trošku složitější „to-do-list“. Ale já, co dělám na sebe, co mám projekty, co mají tasky a tohleto...tak já dělám na Kanbanu, protože já to mám prostě tam napsáno, kdy to má být hotový a podle toho i dělám. Mám všechno naházené v jednom boardu, oštitkováné, do kterého projekty patří, a jedu. Co jsme měli třeba klukama v XXX, tak jsme normálně měli 14denní sprinty, klasické iterace, plánování, všechno...to jsme tam měli a...jako řešili jsme to. A já jsem nic jiného nehledal, mně stačily tyhle ty dvě, a jestli...přiučí někdo k budoucnu [smích].

### **Problémy při implementaci**

#### 8. Byl/a jste zaměstnán ve firmě během implementace agilních metod?

- a. Setkal/a jste se při implementaci agilu s nějakými problémy?
- b. Pokud ano, s jakými? Jak jste jim čelil/a?

Byl...vlastě jsem byl u zrodu Scrumu v té první firmě, v té druhé firmě, a i ve třetí.

Když to řeknu u té první, což byl korpo...velký startup, spousta lidí, a tam jsem u toho byl jako vývojář...není to někdo, kdo to zavádí...takže tam ten přechod byl celkem v pohodě. Když jsme se vlastně k tomu dostali, nějak jsme to nastavili, fungovali jsme. Pak jsem se dostal na pozici Scrum Mastera, což asi nebylo úplně něco pro mě, protože já mám docela svérázný způsob vedení a tohle to...samozřejmě, v určitých mezích...abysme se někam nedostali. A tam začal být problém, kdy můj nadřízený s tím konkrétně nesouhlasil. Takže my jsme měli pomalé sprinty, kdy jsem se snažil to zrychlit takovou tou lehčí kritikou lidí, co dělali a nedělali, tak jsem za to byl trestaný já. Pak jsme samozřejmě byli pomalí, tak to taký jsem odnesl já, takže nakonec tam to bylo strašný. Což ale nesouvisí s přechodem, tam šlo spíš o tom, že se ty lidé na to zvykali, že tam to bylo většinou noví lidé a tak.

Ve druhé firmě jsme na to přešli...tam jsme šli od klasického to-do-listu přes Kanban až ke Scrumu, kde Kanban vlastně byl úplně v pohodě, tam se nic neřešilo. U Scrumu jsme měli problém s grafikama, kteří úplně neumí fungovat na systému sprintů. Prostě řeknou...nemám múzu a 14 dní koukám do počítače...což vedlo docela k rozbrojům a takovým to věcem, nebudu to dál rozvádět. Každopádně, tam byl problém s lidmi, kteří úplně nepochopili ten princip, a ani nebyli schopní tak pracovat, což asi byla i moje chyba, že jsem to po nich chtěl.

A ve třetí firmě...no, ve druhé firmě jsem to zaváděl vlastně já, ve třetí jsem to taky zaváděl já a šli jsme stejným stylem. Začali jsme nějakým to-do-listem, Kanban, Scrum. Ten Scrum tam nevyšel, protože tam byl velký šéf, kterému se nelíbila rychlost té práce. Prostě nebyl schopný ani ochotný to akceptovat, že ten začátek prostě trvá.

Což jsou takové ty největší problémy s implementací.

#### 9. Víte, nakolik je těžké, nebo snadné sestavit tým, který by vyhovoval agilním principům?

Nad touto otázkou jsem se pousmál, když jsem to četl, nic ve zlém. Není to otázka, jestli je to těžké, nebo snadné, ono je otázka, jestli jsou ty lidé ochotní v tom pracovat. Pokud jsou ty lidé *všichni* ochotní v tom pracovat, tak je to snadný. Pokud jediný z deseti lidí není ochotný, tak je to neskutečně těžké a blíží se to k nemožnému. Tam prostě stačí jeden sabotér...je to docela nesmyslné slovo, ale jeden kdo prostě...i kdyby třeba podvědomě nechtěl s tím pracovat, tak to prostě kazí a kazí to všem a tohle to prostě stojí a padá na tom. Už, když je to zavedené ve firmě nebo tohleto, tak ten vývoj určitého produktu, projektu závisí na tomhle. Jestli to prostě spadne nebo nespadne, to lidé musí buď všichni chtít – nebo dělat, že chtějí [smích] – nebo to prostě jde do kytek. A pak se dostaneme k anarchii a Scrum se tou anarchií dělat nedá. Že historická anarchie k ničemu nevede, toto jen to potvrzuje.

10. Pokud někdo do týmu nezapadá (jsou v týmu interpersonální konflikty), jak to řešíte?

Tak to bych řešil pár fackama, no [smích], ale samozřejmě to nejde. My jsme to řešili...já jsem tohle to řešil dvakrát, kde беру celý tým jako Product Ownera, Scrum Mastera a nějaké vývojáře. Vlastně ten první problém jsem zmínil. Ten Product Owner byl takový, že se to prakticky nedalo řešit. V podstatě, co jsme řekli, co jsem řekl, co jsem se snažil tam dostat do toho týmu, tak ve výsledku se skončilo tak, že to bylo ne sabotováno, ale prostě zničený, sestřelený, že to se vlastně dělat nebude. Takže neměl jsem tu volnost...ne volnost, ne, to je blbé slovo...neměl jsem jakoby tu možnost něco s tím udělat. Protože když to kazí někdo nad tím, tak ten tým jde celý do kytek. To jsme vlastně řešili i v tom druhém případě, takže i v třetím, co jsem říkal prvé, tak tam zase to bylo...ne Product Owner, to byl nějaký šéf startupu, tam byla podobná situace. Když on vás nenechá, tak vy to neuděláte. A vrátím se ještě k druhému, tam s těma klukama...jak jsem mluvil o graficích...tak tam to bylo už opravdu na facky. Řešili jsme to tak, že grafiky jsme oddělili od vývoje, měli vlastní scrumový tým a jeli na měsíčních sprintech. Ale výsledek nebyl žádný z toho. Takže tak jsme to řešili. Takže pokud je tam nějaké zkřížené...nějaké oddělení, tak je nutné to rozdělit, tak bych to řekl, tak jsme to řešili, zkusili jsme to. Pokud ne, prostě v nejhorším případě pustit to. To se svým způsobem stálo, ale teďka to nebudeme rozebírat. Takže asi tak.

### **Dodržování principů agilu a praxe**

11. Je u Vás používán princip kolektivní viny (za chyby zodpovídají všichni členové týmu)?

Bylo. Ale ve výsledku...no, v těch dvou posledních případech to bylo. To odnesli všichni...v tom prvním případě to odnesl Scrum Master, takže já. Takže já jsem se tam těch lidí dost zastával. Každopádně princip kolektivní viny v tomhle tom kontextu...záleží, jak myslíte ty chyby. Jestli v průběhu vývoje nastane nějaká chyba jakoby ve vývoji tohoto produktu, tak tadydle neexistuje něco jako vina. Protože Scrum jako takový v základu by měl zamezit, v nějakém tom kroku by měl chybu najít a vrátit se a znovu se jede. Prostě další iterace a jedeme. Samozřejmě to tak není, jo, pak se hledá nějaký „obětní beranek“. Pokud myslíte chyby jako té, co se dějí celkově v implementaci tohoto Scrumu nebo celkově tohoto chování týmu...tak tam by ta kolektivní vina neměla být určitě, tam za to může nějaký konkrétní člověk, kdo to boří. Samozřejmě když tam jsou dva, tak asi to nazvu kolektivní vina, protože to rozboří všechno...jak jsem řekl dřív, tady možná není od věci toho, kdo to kazí, tak prostě pustit.

12. Jak byste popsal/a vztah Vašeho týmu se zákazníkem?

Jo, na to jsem se připravoval, tak i odpovím. Takhle...klient v mé současné situaci se ke Scrumu nedostane. Když jsme to řešili poprvé, tak tam vlastně ten vztah se zákazníkem byl takový, že ten zákazník vymyslel nějakou blbost a kvůli tomu se rozdrobil sprint. To je jednoduchý. Vlastně zákazník si vymyslel něco, a celý sprint šel do kopru, protože se to mělo rychle udělat pro toho konkrétního zákazníka. To je něco, s čím jsem prostě nedokázal existovat, a to je jeden z důvodů našeho rozbroje s Product Ownerem. To je zrovna takový konkrétní příklad. Vlastně v XXX se to nestálo, a jelikož se ve třetí firmě jsme žádného zákazníka ještě neměli, tak tam...nevím. Každopádně ti zákazníci jako takoví...je to nezajímá. Oni se zeptají, jestli to máte – v pohodě, – a jaké to má rozsáhlé následky, je jim to jedno. Je prostě zajímavá, jestli za své peníze dostanou to, co chtějí. To řeknu přesně: za ty peníze dostanou to, co chtějí, plus 200 procent věcí navíc, takže...za stejné peníze, samozřejmě. [...] Tohle to je krásně na Scrumu, kde na konci každého sprintu by měl být nějaký release, popřípadě nějaká ukázka něčeho, aby se na to ten zákazník podíval. Ale pro ně je to prostě nějaký release, build, popřípadě něco takového, a v 90 procentech je nezajímá, jak se k tomu docílilo. Prostě už jsou zvyklí, že jednou za 14 dní uvidí nějakou změnu, i kdyby...já jsem zažil, kdy ten zákazník byl jakoby nadšený z toho, že viděl změnu ze žluté na červenou. Takže takhle to je. Samozřejmě fidelně se to stane, že prostě se dělá rychle release v průběhu sprintu, a znova – kopr, všechno špatně.

13. Jak často se stává, že Product Owner nebo Scrum Master má pracovat s více než jediným týmem? Může to způsobit problémy v projektu?

Může. Ono by se to správně stát nemělo. Měl by být prostě scrumový tým, ten se skládá z produktáka...z Product Ownera, ze Scrum Mastera a nějakých vývojářů. Nebo grafiků, čehokoliv. To je prostě ideální stav...nějaký black box, a ty si vyšívají na vlastním písččku. Prostě...jenom dělají to své, to, co potřebují, aby se to odbavilo. Zažil jsem jak Product Ownera, který měl nějakou část týmů, tak i Scrum Mastera, který měl nějaké týmy, a dokonce jsem zažil i to, že Product Owner měl třeba šest týmů pod sebou, Scrum Master jeden, tři, druhý – další tři, a takhle se to celé míchalo a byla to jenom cesta do pekel. Zaprvé, lidé na to neměli ani pak čas, pomáhat těm týmům, kdy ten Scrum Master nestíhal prostě...jako když se stará o tři týmy, a...i kdyby každý den za nim přišel jeden, tak je to šíleně moc, nemá čas na nic jiného, nemá na to čas vůbec připravovat, plánovat...nemá čas na to připravovat sprinty, nemá čas vůbec nějak diskutovat nějaké fičury a tohle to. Takže ano – může to způsobit problémy.

14. Vyžadujete od PO a SM nějaké certifikáty? Pokud ano, jaké?

Já určitě ne [smích]. Já jsem...to většinou dělám já, jo [Scrum Mastera]. Vím o tom, že tam byli kluci, co nějaké certifikáty měli, ale že by se vyžadovaly, to ne. Ono vlastně...tenkrát to

bylo tak, že ten Scrum Master a Product Owner byl za odměnu, ale šlo spíš o tom, jak dlouho jsme tam tenkrát byli. Já jsem byl v té firmě už od začátku skoro, takže jsem se dostal na Scrum Mastera, já jsem si myslil, že se polepší a to. Takže...nevyžadovali to nikdy, co vím, a že firmy, které znám, to taky nevyžadují. K čemu to je? Upřímně, je to dvoudenní školení, na konci kterého dostanete takový kus papíru. [...] To je prostě duch moderní doby – čím víc člověk má papírů s nějakým razítkem, tím jako je lepší, což je blbost. [...] Tady je prostě spousta lidí, kteří se tím honosí, jak Scrum je super. Nemám proti tomu nic. Ale jak to všechno učí...20 tisíc prostě na hlavu. Já jsem viděl nabídku od člověka, co dělá nějakého řádového testera v Avastu a založil si na tohle firmu. *Tester*. Člověk, co prostě kliká, jestli něco funguje – tak chce přednášet o Scrumu. Mozek to nedokáže pochopit. Nemyslím to moc namy...no, trochu jo, – namyšleně, ale prostě ty lidé jsou blázní! Nejhorší jsou teda ty, kdo jim to zaplatí. I tak spoustu pěkných keců, ty firmy to pak udělají špatně. A tady máte tohleto školení a pak pomoc s implementací...aaa, asi se dá na to pěkně vydělat, já jsem si vybral blbej obor. A, třeba se to ještě změní.

#### 15. Na kolika agilních projektech může zároveň pracovat jeden člověk?

Myslíte na projektech v rámci firmy, nebo úplně obecně, globálně? Je to dost stejný. Já jsem toho názoru, že ten člověk může pracovat na několika projektech. Jenom ty projekty musí mít správně rozhozené do toho sprintu. Což, samozřejmě...když je to v rámci jedné firmy, je relativně snadný. Samozřejmě v jednom týmu ještě, jo. Kdyby ono bylo jakoby napříč týmama, dá se to, ale zblázní cestou ten člověk, což vlastně souvisí s tím, kdyby to bylo globálně. To už by pak existovalo. Když jsme zakládali XXX, tak jsem vlastně měl nějaké dva projekty v práci a pak ještě jeden ve svém volném čase. Takže jsem měl tři projekty – ve dvou firmách, ve dvou týmech, a fungovalo to. Prostě jsem tomu ten sprint přizpůsobil. Věděl jsem, že od 9 do 5 jsem v práci, a moje pracovní doba je od 6 do 8 rána, a pak od 6 do 8 večer. Takže 4 hodiny denně věnuju svým projektům a 8 hodin jsem v práci, což je s přepočtem na hodiny ten scrumový den. Takže ano, může pracovat jeden člověk na více projektech. Ať už napříč firmama, týmama, všelijak. Samozřejmě když je to v jednom týmu, jednom boardu jakoby v jednom sprintu víc projektů, tak – úplně ideálně, protože „přepínání“ mezi projektama...asi někomu to vadí, ale ve výsledku je ta práce pořád stejná. Člověk to zvládne, i když může z toho zbláznit. To zaleží, jak to má s firmou a jestli má do toho započtené nějaké své projekty a tak. Opravdu zaleží, jak člověk to konkrétně nastaví. Je to o tom člověku. Takže může pracovat, ale když to nezvládne, nic se nestane.

#### 16. Popište prosím, jak probíhá komunikace mezi členy týmu.

To myslíte, jak se baví nebo jak si nadávají? Baví se po telefonu nebo po Slacku. Ale jinak ta komunikace v tom scrumovém týmu musí být prostě suprová. Ty lidé nesmí jít proti sobě, protože to prostě boří všechno. To bude znět tak hodně klišoidně, jo, ale musí se věřit. Musí vědět, co od toho druhého čekat, a popřípadě ho podpořit, pomoci mu. Samozřejmě za cenu, že se neutrhne od toho svého. Každopádně ta komunikace by měla být fajn. My jsme to tak většinou měli. Většinou to kazí lidé z managementu, takže Product Owner a víc. Tam ta komunikace...no, ten Product Owner do toho týmu jakoby patří, ale to prostě nesmí být člověk, kterého zajímají čísla. To musí být člověk, kterého zajímá ten vývoj. Takže ta komunikace by měla být dobrá, u nás většinou to byla dobrá, někdy jsme samozřejmě tak nějak poštěkali, ale tak. Víte, když to děláte ve svém volnu, vlastně za své peníze, jo, tak se člověk poštěká, i „proč to není hotový“ a „to by mělo tam být“ a tohleto. Ale to k tomu patří. Ale jinak jsme používali Slack, takový komunikační nástroj, tam je to super. Ale to asi není přímo odpověď na tuto otázku, jo [smích]. [...] Jira je scrumový nástroj, Slack je čistě komunikační služba. Jira je hrozný paskvil, to nepoužívejte. Targetprocess je lepší. Rozšířená je ta Jira, druhá je...Asana, myslím, že se to jmenuje... A suverénně nejlepší – a nejdražší teda [smích] – je ten Targetprocess. Ten je cenově úplně někde jinde. Na druhou stranu, má tolik služeb, že když to používáte rozumně, tak se dá to používat i zadarmo. Takže...je to super nástroj. Používali jsme ho ve všech firmách.

17. Používáte nějaký software k plánování sprintů, meetingů, pro backlogy atd.? Pokud ano, uveďte název.

Aha, ale tady to máte... Takže software pro plánování sprintů...[smích]. Takže jak jsem říkal: Jira je hrozný paskvil, Asanu neznám, jenom ze slechu. Dá se použít Trello, ale na Scrum se tam musí použít plugin, a na freečkovém Trelu můžete použít jeden plugin, takže ten jsem nezkoušel, a suverénně lepší je Targetprocess. To dělají v Bělorusku, shodou okolností. [...] Je to úplně úžasný nástroj. Takových funkcí, co to má, takovou možnost k jakoby „customovatelnosti“, různých nastavení a tohleto, – upravit dle obrazu svému – to je prostě paráda. Stojí to 25 dolarů měsíčně na člověka, což je pro startupy šílená cena. Nám tenkrát nabízeli slevu, že nám to dají za polovic. A existují u nich freečková verze, kdy tam máte tisíc entit zadarmo, jinak všechno je stejně jako u plné verze. Dřív měli pro pět uživatelů, to jsme používali my, teď je tam tisíc entit. Entita je úplně všechno. Normální klasická kartička... tasková, bugová, – toto je všechno entita, vlastně uživatele entita. Tisíc entit tam máte, jo, a málem ti to stačí. Oni se pak odmazávají, takže když máte...já nevím...na jeden sprint 40 storek, tak vám tam to vydrží 20 sprintů, 30 skoro no. 20, když budeme počítat, že komentáře a tohle to taky jsou entity. Takže ten je nejlepší. [...] Když vezmeme Slack, ten má 10 tisíc práv

a pak to odmazává. To je úplně to samý. Prostě pro malé týmy to stačí. Velké korporáty, ty za to zaplatí, tím to nevadí. Paradox, to se prostě promítne v cenách.

18. Jakou délku sprintu obvykle máte?

My jsme začínali na týdenních, ale tam se to nestíhalo testovat – co se týká nějaké procesu, kdy jsme měli dvoufázové testování, – takže jsme přešli na 14denní sprinty. A jako jsem říkal předtím, tak vlastně v XXX grafici měli měsíční a vývoj měl 14denní. 14 dní je...ono zaleží, na jakém projektu pracujete. Když děláte víc malých projektů, tak potřebujete mít týdenní sprinty. Když děláte jeden obrovský projekt, který se vyvíjí čtyři roky, tak klidně můžete mít měsíční sprinty, tam se nic nestane.

19. Pokud v průběhu sprintu nějaký úkol není splněn, co se s úkolem dále stane? Jak často se Vám to stává?

Oh...tak stává se to častěji, než by mělo. Ve všech firmách, co jsem byl, to se prostě stálo a zůstalo se v jakémkoliv stavu, ať je otevřený nebo v testování. Vždycky se to stalo, jednoduše se to přesunulo do dalšího sprintu. Padla velocita, a byl problém na střeše, no. Každopádně, tohle to se určitě děje na začátku, když prostě nevíte, kolik toho máte naplánované. Po...já nevím...po 10–20 sprintech by se to stávat moc nemělo, nebo jak je to dneska...víte, kolik toho stihnete, kolik toho uděláte, a už jsou ty lidé tak zkušený, že dokážou to přesně nadhodnotit a vědět, že všechno by mohlo projít.

20. Jaká je střední délka plánovacích meetingů?

Ježišmarjá...já tuhle tu metriku tak nemám rád. Když to řeknu úplně střední, tak bývá hodina a půl. Taková střední délka. Hodina je málo, dvě hodiny je moc. Takže já nevím, co s tím mediánem nebo průměrem...[smích]. Takže hodina a půl, no.

21. Daily stand-up by měl ideálně být každé ráno ve stejnou dobu, ve stejném místě, ve stoje a maximálně 15 minut. Jak to máte Vy?

Ideálně je ještě pořádně se u toho hýbat [smích]. Zaleží na velikosti toho týmu. [...] V té první firmě, což byla velká firma...tam byl vlastně 10členný tým a ten stand-up trval 10–15 minut maximálně. Ideálně jsme byli hotoví za pět. Samozřejmě, ten princip, jo. Ještě pět minut před tím všichni slézali, pět minut potom se ještě všichni nadávali [smích]. Ale ten vlastní stand-up byl pět, pět-deset minut. Ideálně. Tam nemusíte říkat, jak se máte nebo něco. Jen řekněte: „Hej, já jsem včera udělat tohleto, tohleto a tohleto, řešil jsem tohle, tohleto...jak se to řekne slušně...nepovedlo a dneska na tom budu pokračovat“. Takové věci udělají lidé hned, tak máte

na člověka 20 vteřin...comohl stopnout, a je to hotový. Ve třetí firmě jsme ho neměli, a ve druhé jsme měli stand-up vlastně po Slacku ráno. Takže jsme u toho tam ani nebyli, prostě kluci napsali, co udělali za ten den předtím...většinou to bylo „včera jsem neudělal nic“, ale víte, jak to funguje. Když zakládáte firmu, buď chtějí, nebo nechtějí dělat, jo. Takže takhle jsme měli stand-upy. Slack-upy. No, já to tak nemyslím. Každopádně jsme to napsali – co se dělo, co se bude dít, a bylo. Ale fungovalo to. Chvilí. Ono je hlavně fajn o tom, že v rámci toho týmu víte, nad čím konkrétně ten člověk pracuje. I když si vezmu ten náš obor, vývoj nějakého softwaru, vy pak vidíte prostě, s čím má problémy. Když vám čtyřikrát na stand-upu řekne, že teď řeší průniky polí nebo něco takového, tak asi víte, že s tím má problém, takže můžete třeba zařídit školení nebo ho profackovat a je to jedno. Takže fakt ne, co mi přijde hodně, je desetičlenní tým. Samozřejmě by víc ten scrumový tým mít neměl. Když máte 50 lidí ve scrumovém týmu, tak se můžete dostat k tomu, že jste u vodopádu – svým způsobem, jo. Prostě v chaosu.

## **Výsledky implementaci**

22 Dá se říct, že díky používání agilu se po čase výsledky práce ve Vašem týmu zlepšily?

Zlepšily se. Určitě se zlepšily, ve všech těch firmách. Jenom to prostě chtělo čas. Jde hlavně o to, že po každém sprintu máte co ukázat. To jedna z věcí, co ty zákazníci chtějí, tak to je prostě. Výsledky se zlepšily – po čase, samozřejmě. Zhoršily se po začátku: první čtyři sprinty – prostě šílený propad. Tam ve výsledku se nic neudělalo. [...] Záleží, jak dlouhé máte ty sprinty. Jestli máte týdenní, tak jste za dva měsíce v pořádku. Pokud máte 14denní sprinty, tak to může trvat klidně déle, protože vy si zvykáte na tu velocity. Třeba jde udělat to, že začnete na týdenních sprintech, abyste se do toho dostala, a pak přejdete na delší sprinty, aby z toho jakoby něco vypadlo. Tady je opravdu důležité to... tady je spousta lidí proti bodování, jo, a tady je důležité nastavit si tu velocity. A nějaký systém bodování. Zcela nejdůležitější pro toho konkrétního člověka je, aby si nemyslel, že jeden bod je jedna hodina. To je cesta do pekel. A jakmile se tyto body přepočítává na hodiny, tak se vám stane to, že vám tam ty lidé budou chodit jako na píchačky. Že vidí prostě, že musí udělat 8 bodů denně, a udělají těch 8 bodů, a když za čtyři hodinky budou hotoví, tak čtyři hodiny budou pak...já nevím, koukat na YouTube. A je to v pytli. A už se nepřijde na to, že byl blbě ohodnocený ten sprint, celý, na což by se mělo přijít. Aby to bylo co nepřesnější, a potom se dá to ve výsledku zjistit. Ano, je to jeden bod, je to 40 minut, celý ten projekt nás stál tisíc bodů, což je 40 tisíc minut, já to teď nespočtu...a dostali jsme za to zapláceno a stalo nás to tolik a tolik peněz, a dostali jsme



zapláceno třikrát tolik...no, tak jsme na tom vydělali hrozně peněz. A víme to přesně, což zlepšil jak ten vývoj, tak i celou tu strukturu v té firmě tím, že prostě vydělávají ty peníze. Protože lidé to dělají pro peníze. Kdo tvrdí, že ne, tak zaprvé je pokrytec, zadruhé je blbec.

### 23 Mělo zavedení agilu dopad na rozsah dokumentace k projektům?

To raději přeskočíme, já žádnou nepíšu [smích]. Nemělo. Ta dokumentace...jakoby správně když máte nějakou storku za úkol a tohleto, tak ta dokumentace by měla být součástí toho. Takže by ta storka měla s tím počítat, jestli se dělá ta dokumentace, testy a tydlety věcí. Správně by to nemělo být. Když jsem říkal, že se tam změnila ta barva, tak dobře, tak to ohodnotíte prostě na bod a někdo ještě řekne, že tam přeje tu dokumentaci, ty testy a tohle to, tak dobře, tak to už jsou dva body. Tak napíšete to, změňte barvu prostě jednoduše, a už na to napíšete osm testů, jestli je ta barva třeba dobře viditelná. Nevím, jak by se to testovalo [smích], ale je to jedno. A k tomu napsat dokumentaci. I když to řeknu blbě...pokud se v barvě používají složky červené a zelené a modré, jo. Proč? A napíšete to. A najednou z úplně primitivního úkolu máte trochu rozsáhlejší. Takže dopad na rozsah dokumentace...to záleží na tom, jestli si ty lidé tu dokumentaci přáli nebo ne. Ten Scrum jako takový nemá žádný vliv – z mého pohledu teda samozřejmě, jo. [...] Ke každému projektu dokumentace by měla být. V dnešní době...zase, když se budu prostě bavit o softwaru, tak je dneska spousta nástrojů, které udělají dokumentaci s komentáři v kódu. Dneska nedělat to je spíš...no, já mám spoustu nezdokumentovaných projektů, jo. A když jsem se teďka k nim vracel po nějaké době, protože u nás dneska v Čechách je přechod na GDPR, což je kapitola sama pro sebe... Každopádně, jsem se dostal ke starším projektům a tam jsem se musel nafackovat to, protože nevěděl, co s tím dělat. Takže nemělo by to mít žádný vliv prostě. Pokud ta firma dělala všechno správně před zavedením, že komentovala, tak budou komentovat i potom, pokud nekomentovali, tak nebudou komentovat ani potom. Když se dají do toho procesu scrumového, mají nějaký vývoj a mají stejně i dokumentaci, tak ok, ale pokud to někdo zavede, tak se z toho zblázní.

### 24 Bylo podle Vašeho názoru zavedení agilu přínosným pro Váš podnik?

Pokud vezmeme přínos finanční, tak svým způsobem jo. Pokud berete přínos jakoby pro ty lidi, že se tam vyvíjeli něco, naučili se, zlepšil se proces a tohleto, tak určitě. Určitě se zlepšil proces toho vývoje, projektů, určitě se zlepšila komunikace, určitě byly lepší výsledky z toho.

## Hodnocení a doporučení

### 25 Jaké pozitivní dopady může mít implementace agilních metod v podniku? Uveďte příklad ze svých osobních zkušeností z praxe.

Tak to jsem říkal před chvílí. Určitě se zlepšila ta výkonnost...ne výkonnost, to je blbé slovo...určitě se zlepšily výsledky toho, jak to vypadalo nebo nevypadalo. Lepší styl práce určitě byl lepší...už jsem řekl čtyřikrát „lepší“, co? Jak říkám, prostě hezčí kód, rychlejší práce, lepší výsledky. A hlavně – hmatatelné výsledky. Že se dva měsíce nedělají nějaké blbosti.

26 Jaké negativní dopady může mít implementace agilních metod v podniku? Jaká negativa jste ve své praxi zažil?

Jo, negativní dopady... To jsem taky říkal několikrát. Pomalý start. Takže té starty byly těžké, proto těm lidem to vadilo. Že se vlastně nic neděje pořádně. Další dost takový negativní dopad...když se na to prostě někdo vybodnul, vykašlal, tak to bylo takový...dost na nože. Ale je to spíš následek toho všeho. Ale to je negativní dopad, prostě několikrát jsme se rozhádali s klukama, se kterými chodíme na pivo, takže já bych to taky vzal za negativní dopad, prostě „proč se to neudělalo“, „co se to udělalo“ a „proč ses na to vykašlal“ a tak.

27 Co byste doporučoval/a lidem, kteří chtějí dané metodiky implementovat?

Oh...je to na více úrovních. Pokud se bavíme o nějakém velkém CEO nebo něco takového, tak ať do toho nekecá. Pokud se bavíme o nějakém šéfovi vývoje nebo něco takového, tak ať je klidný. A když už se bavíme konkrétně o tom týmu, tak to, jak bylo v nějakém filmu – musí vydržet. Ten začátek bude opravdu těžkej. Nesmí se nechat vyprovokovat s tím, že by jim to někdo chtěl zkazit. Musí se prostě držet. Hele, tadyhle je sprint a teď ho musíme prostě udělat. Jako samozřejmě stane se někdy to, že máte půlku sprintu a jste hotoví. A pak se stane to, že do toho narazí jiné věci, což je konec světa. Ať se ty lidé půjdou zabit radši místo toho. Ať jim ten sprint neboří. Zjistí, že „zvládli jsme to brutálně rychle, ale nic nám už tam nepřidáte“. Ten graf [burn down] pak dělá tohle [...] a to se nikdy nesmí stát. Nikdy nesmí nahoru. Ten graf ideálně má jít takhle. Samozřejmě jde takhle [...] protože máte víkendy a takovýto věci. A samozřejmě ne každý den se udělá všechno. Ale měla by být taková ta krásná úhlopříčka. Takže tak. Jestli fakt mám doporučovat někomu něco, tak prostě vydržet. Opravdu stojí to za to.

28 Chybělo Vám něco v našem dotazníku? Chtěl/a byste jej nějak doplnit?

Ptáte se na sprinty jako takové, na tu časovou jednotku, to je jasný. Ale jsou firmy, které to bodování nepočítají, nepoužívají, ale jsou firmy, které za tím bodováním stojí. Já jsem ten druhý případ. Zrovna bodování je takové, že každá storka, nebo bug, samozřejmě, tak se ohodnotí nějakým počtem bodů. Nevím, jestli jste se o tom slyšela. Vlastně v základu je to 1, 2, 3, 5, 8, 13 21 a nekonečno. Existuje takové pravidlo...no, jak pravidlo. Dřív se to počítalo tak, že jeden bod je jedna hodina, to už jsem říkal, že je to blbost. To takhle nejde. Teďka...zpravidla se to

počítá tak, že by user-story neměla bodově přesáhnout jeden den. Což samozřejmě podle velocity víte, kolik je. Zůstane vám to na osmičce, takových těch osm hodin, osm bodů, jo. Víím, že teď vyvracím to, co jsem řekl, ale jakoby...dobře se to počítá potom. Takže dejme tomu, že nám z velocity vyšlo, že opravdu jeden bod je jedna hodina. Což je takový krásný stav storek, ze kterého by každý projekták měl radost, protože může vykazovat hodiny. Takže praktické by žádná storka neměla přesáhnout těch 8 bodů. Pokud se to stane, měla by se roztrhnout – na dvě. Kdyby vyšla na 15, tak by to bylo 8 a 8 třeba. Aby nepřesáhla těch 8 bodů. Aby se obecně nepřesáhla ten sprint, nebo den toho sprintu. Když v 14denních sprintech máte 10 dnů, ono se dá spočítat, ale tím se teď nebudeme zabývat. Takže tak se to dělá. Vlastně při plánování se děje to, že se každá storka projde a ohodnotí to ten tým. Používají se na to scrumové kartičky, kde jsou ty body vytištěny a jakoby to ty lidé hodí na stůl. [...] To se jmenuje „planning poker“. Ono je moc obsáhlé, a já mám radši tu 21 než 20, ale je to jedno, ten princip je stejný, ta vysoká čísla by tam být neměla. Samozřejmě se může stát, že jste v takovém jakoby super týmu, že máte 20 bodů na den. Stane se to, proč by ne? Ale na to pak jsou tyhle ty čísla, jo. Normálně se to nestává. Já jsem to sám nezažil, měli jsme to na nekonečno. Ale půlku a nulu, to jsme nepoužívali. Nula je prostě nic, a půlka – no, to je na prd. Jednička je nejnižší jednotka. Každopádně je to ono, „planning poker“ se to jmenuje, a lidé prostě otočí té karty, položí na stůl, a pak se řekne, že všichni mají ohodnoceno, otočí se a hledá se nějaký průnik. Zajímavé je to, že různé firmy té průniky dělají jinak. Používali i medián teda [smích], to jsem taky viděl. Používalo se i to, že se to i průměrovalo, že se nejhorší a nejlepší vyhodilo a udělal se průměr z těch zbylých, takový sportovní. Takže spousta věcí se tímto stylem dělalo. Každopádně při plánování víte, že vám ta storka sebere 4–5 bodů, tak je to skoro půldne...a můžete. A víte, že takovýdle storky, kde...že uděláte 40 za sprint...no, dobře, uděláte 200 za sprint těch bodů...ve velkém týmu, tak prostě takových storek se tam můžete naskládat 40. A pak z toho vypočítává velocity toho sprintu, a vy víte, kolik jste z toho udělali, a zjistíte, že máte za poslední sprint 200 bodů, fajn, tak 200 bodů si naplánujeme znova. Zjistíte, že jste udělali 150 – tak skvěle, no, že 50 nám spadlo z minulého sprintu a teďka prostě dalších 100, že můžeme je naskládat z nových věcí. Samozřejmě má smysl jakoby po těch bodech chtít víc, když to prostě nezvládnou. A toto je systém těch bodů. S tím samozřejmě, že existuje něco, čemu se říká „burn down“, což je ten graf, co jsem ukazoval. On má ideálně vypadat takhle. Na začátku sprintu máte plný počet a na konci sprintu máte prázdno. Musíte prostě takhle sjet. Většinou první den neudělíte nic. Ono se začíná den po plánování, jo [...] V praxi to bývá tak, že na konci sprintu, poslední tři dny máte takovýhle propad dolů a pak se jede. A z toho jste pak schopní vypočítat, jak ty lidé pracovali. To opravdu zní úplně šíleně, fakt prostě pracovníci a je

naplánováno všechno a to, ale není to tak. Ono opravdu pomůže a opravdu ty lidé zaprvé se naučí pracovat jako tým, pořádně a zrychlí to ten vývoj brutálně. Není to tak, že musíte pak dva roky pracovat za 200 bodů, jo. Sice se dopracujete k tomu, jak se ty lidé vyvíjí. Bavíme se pořád o tom, že ty fičury jsou stejně velké a můžete udělat třeba 250 za rok. Jakoby za sprint, ale se naučí to dělat za rok a furt se zlepšují. Vy pak zjistíte, kolik to stojí ta firma. A ruku na srdce, už jsem to řekl dneska několikrát – ty peníze jsou až na prvním místě, že jo. [...] Jinak jste zabrala v podstatě všechno, akorát ty body teda. To jsem se divil, když jsem to pročítal. Ale zase chápu, že spousta firem na to nechce fungovat. [...] Pak se dostáváte blíž ke Kanbanu. A chaosu. Protože se stává, že ten produkták tam nahází miliardy věcí a musí se prostě stihnout. A jestli je na Scrumu něco krásného, takže on v principu nepočítá se s přesčasem. Na což dneska lidí chytnete, že jo. Protože nikdo nechce dělat zadarmo. Takže tohle to tomu zamezí. Ne tomu, že se někdo zblázní, že se mu klient zblázní, že potřebuje mít tohle to hotový a dá tam něco, co lidí zaboha nemůžou stihnout. A je to pryč, celý koncept toho je rozdrobený a rozšlapaný. A body...no, já to prostě mám rád. Já to mám i ve svém Kanbanu svým způsobem, protože vím, kolik za týden toho udělám a nemůžu si naplánovat 14denní práci na jeden týden a nikomu nemůžu slíbit, že to v pátek bude mít hotový, když to prostě mít nebudu.

## Příloha 5. Kopie e-mailu R1 z 8.12.2017

Птн, 8 Дек 2017 18:57

RE: Otázky k rozhovoru (agile)

От: ...

Кому: Sofya Garmash <ikilumi@mail.ru>

Копии: ...

Dobrý den vám všem,

Když jsem ještě přemýšlel nad vašimi otázkami, tak mě napadla jedna oblast, která v dotazníku prozatím není - jak se daří jiným firmám při transformaci z waterfallu do agilu realizovat opravdu plnohodný agile, kdy v agile týmech naplno fungují zodpovědní Product Owners, kteří opravdu vlastní produktový backlog a určují prioritu položek v něm. Často se totiž v praxi setkávám s agilem „jako naoko“, kdy se waterfallem připraví analýza, která se jde tzv. agilně implementovat. Tento nešvar je možná způsoben tím, že změna na agile potřebuje transformaci celé organizace, nejen IT a to je často problém.

Proto posílám tuto svoji úvahu pro inspiraci. Bylo by možné z toho vydestilovat otázky typu – „vlastní product owner produktový backlog?“ , „je schopen určovat priority feature?“, apod ....

Přeji vám hodně úspěchů do dalšího snažení,

S pozdravem

...